REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université SAAD DAHLAB Blida 1

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

Réalisé par :

Mederbel Sofiane

Mehal Anis

Thème :

Conception et développement d'une plateforme de gestion de données médicales basée sur la technologie blockchain pour les applications e-santé

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

Soutenu le 14/07/2021 Devant le jury composé de :

Mme.Boustia Narimen	Présidente
Mme.Ghebghoub Yasmine	Examinatrice
Mme.AROUSSI Sana	Promotrice
Mr.Khemissa hamza	Encadreur
Mr.Derki Mohamed Saddek	Encadreur

Promotion : 2020/2021

Remerciement

Avant toute chose, on tient à remercier Allah, le tout puissant, pour nous avoir donné la force et la patience d'achever ce travail.

Nos plus sincère remerciements pour notre Promotrice Madame Aroussi Sanaa pour la confiance que vous nous avez accordée, et les conseils et remarques que vous nous avez donner.

Nous tenant à remercie tout particulièrement Monsieur Derki Mohamed Saddek et Monsieur Khemissa hamza attaché de recherche à CERIST et promoteur de thèse, de nous avoir orienté, corrigé notre travail et encouragé. Et Merci aussi pour votre disponibilité. Nous avons acquis beaucoup de connaissance et savoir-faire bénéfique au cours de nos nombreuses discussions.

On aimerait exprimer notre gratitude à tous les chercheurs et spécialistes, Madame Gheghoub, Madame Boustia qui ont pris le temps de discuter des différent problème et solution de notre sujet. Chacun de ces échanges nous a aidé à faire avancer notre projet.

Résumé

La sécurité et la confidentialité des données médicales a toujours été un challenge. En effet, les différentes données des patients telles que les diagnostiques, les ordonnances et les données personnelles sont toutes stockées de façon centralisée, ce qui résulte un niveau de confidentialité non satisfaisant. Pour remédier à ces problèmes, nous proposons une plateforme de gestion de données médicales basée sur la technologie blockchain pour assurer la confidentialité, l'intégrité et la disponibilité. Cette plateforme utilise le cloud IPFS (InterPlanetary File System) pour éviter le problème de stockage des données médicales (volume important) et pour renforcer la sécurité de ses données. De plus, nous proposons aussi d'utiliser un modèle de contrôle d'accès basé sur la méthode de chiffrement à base d'attribut CP-ABE pour chiffrer ces données dans le but est de préserver la vie privée des patients et faciliter l'accès aux données médicales autorisées. Avec le mariage de ses différentes technologies, notre plateforme assure la cohérence, l'intégrité, et la disponibilité et la confidentialité des données médicales.

Mots-clés : E-santé, Confidentialité, Blockchain, Cloud, IPFS, Contrôle d'accès, CP-ABE.

Abstract

The security and confidentiality of medical data has always been a challenge. Indeed, different patient data such as diagnostic prescriptions and personal data are all stored centrally, resulting in an unsatisfactory level of confidentiality. To address these issues, we propose a medical data management platform based on blockchain technology to ensure confidentiality, integrity and availability …. This platform uses the (IPFS cloud) InterPlanetary File System to avoid the problem of medical data storage (large volume) and to strengthen the security of its data. In addition, we also propose to use an access control model based on the CP ABE attribute-based encryption method to encrypt this data in order to preserve patient privacy and facilitate access to authorized medical data. With the marriage of its various technologies, our platform ensures the consistency, integrity, and availability and confidentiality of medical data.

With the marriage of these different technologies, we will ensure consistency, integrity, availability and confidentiality.

Keywords:, e-health, Privacy, Blockchain, Cloud, IPFS, CP-ABE access control.

ملخص

ضمان أمن و سرية البيانات الصحية تشكل دائما تحدي من منظور الأمن المعلوماتي ,من مختلف البيانات الخاصة بالمرضى مثل الأشعة و الوصفات و معلومات الشخصية غالبا ما تكون مخزنة بصفة مركزية , مما يؤدي إلى ضعف مستوى سرية المعلومات, من أجل معالجة هذه المشكلة قمنا بإستعمال تقنية البلوكشين .

نظرا لحجم الكبير للمعلومات في مجال طبي قمنا بإستعمال تكنولوجيا IPFS من أجل حل مشكلة التخزين و زيادة مستوى الأمن هذه المعلومات مخزنة في سحابة IPFS , و بهذا لقد إستعملنا نموذج التحكم في الوصول إلى مستندات بطريقة التشفير المعتمدة على الصفات CP-ABE , لأجل تشفير البيانات , فالهدف من ذلك هو حماية الحياة الخاصة بالمرضى و تسهيل الدخول إلى البيانات الطبية المصرح له

عن طريق دمج هذه التكنولوجيات سوف نساهم في ضمان المصداقية والسلامة والتوفر وسرية البيانات الطبية

الكلمات الرئيسية: الصحة الإلكترونية، الخصوصية، بلوكشاين ، سحابة ، IPFS، CP-ABE ، التحكم في الوصول.

Table des matières :

Liste des figures

Liste des tables

Liste des acronymes

3-DES	Triple Data Encryption Standard
ABAC	Attribut Based Access Control
ABE	Attribute Based Encryption
AC	Autorité de confiance
ACL	Access Control List
ACP	Analyse en composantes principales
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
ARBAC	Administrative Role-Based Access Control
CBAC	Context-Based Access Control
COVID-19	coronavirus disease 2019
CP ABE	Ciphertext-Policy Attribute Based Encryption
DAC	Discretionary Access Control
DAPP	Decentralized application
DDos	*Denial of Service attack*
DES	Data Encryption Standard
DHT	Distributed hash table
DME	Dossiers médicales électroniques
DMP	Dossier medical personnel
DSA	Digital Signature Algorithm

DSE	Dossier de santé électronique
ECC	Elliptic curve cryptography
EHR	Electronic Health record
EVM	Ethereum Virtual Machine
IaaS	infrastructure as a Service
INCITS	International Committee for Information Technology
IoT	Internet-of-Things
IPFS	InterPlanetary File System
IPNS	InterPlanetary Name System
KP-ABE	Key-Policy Attribute Based Encryption
LGoE	leLe graphique logique de la preuve
MAC	Mandatory Access Control
MD4,MD5	Message Digest
MK	Master Key
NIST	National institute of standards and technologie
NPM	Node Package Manager
OMS	Organisation Mondiale de la Santé
P2p	Peer-to-peer
PaaS	platform as a Service
PBFT	Practical Bizantine Fault Tolerance
PFS	Secret de transmission parfait
PK	Public Key

POET	Proof-of-Elapsed-Time
POS	Proof-of-Stake
POW	Proof of Work
RBAC	Role Based Access Control
RDF	Resource Description Framework
REST	Representational state transfer
RFID	Radio frequency identification
RSA	Rivest, Shamir et Adleman (technologie de cryptage à clé publique)
SaaS	Software as a Service
SGBD	Gestion de base de données
SGX	Singapore Exchange
SHA	Secure Hash Algorithm
SK	Secret Key
SSL	Secure Sockets Layer
TCSEC	Trusted Computer System Evaluation Criteria
TIC	Technologie de l'Information et de la Communication
TLS	Transport Layer Security
Web	World Wide Web
Wifi	Wireless Fidelity
WSN	Wireless Sensors Network
XML	eXtensible Markup Language

Introduction Générale

L'E-santé est défini comme l'utilisation des Technologies de l'Information et de la Communication (TIC) pour le transfert et l'échange à distance de données en matière de santé. Les plateformes (ou applications) d'E-santé peuvent apporter énormément au secteur de la santé en Algérie avec le manque avéré de médecins dans la plupart des régions des hauts plateaux ainsi que les régions du sud. En effet, elles peuvent apporter des bénéfices non négligeables, non seulement aux malades mais également à leurs familles en évitant les déplacements pour avis, ou expertise, ou en évitant d'éventuelles erreurs de diagnostic, ou encore de médication [1].

Quand on parle d'E-santé, on doit aussi parler de l'aspect le plus important de ce dernier qui est : l'importance de la sécurité et de la confidentialité qui est l'un des critères clés de notre projet. En effet, une mauvaise implémentation des techniques de sécurités dans l'e-santé peut conduire à la violation des dossiers médicaux, lorsque les entités avec mauvaise attention peuvent avoir un accès complet aux comptes de messagerie, aux messages et aux rapports. Au contraire, si les techniques sécurisées sont implémentées correctement pour l'e-santé, on peut satisfaire toutes les parties prenantes, y compris les patients et les soignants.

Les recherches et les projets existants traitent principalement certains aspects de sécurité dans l'e-santé, telle que ceux basés sur l'IoT (Internet of Thing), basés sur le trafic réseau, et basés sur l'apprentissage automatique. Il est donc nécessaire de faire une recherche qui intègre toutes les approches possibles avec les nouvelles technologies qui utilisent des principes encore jamais abordées dans l'e-santé en Algérie. Nous parlons ici de la technologie blockchain conçu par Satochi Nakamoto en 2008 et utilisé dans la création de bitcoin en 2009. Un Blockchain qui est une chaîne de blocs contenant des lots de transactions, possède certains avantages tels que la sécurité, l'anonymat et l'intégrité des données sans intervention de tiers. Ces avantages en font un choix raisonnable pour gérer les dossiers médicaux des patients. C'est dans ce contexte que s'inscrit le projet notre travail intitulé : « *Conception et développement d'une plateforme de gestion de données médicales basée sur la technologie blockchain pour les applications e-santé »*

Dans ce projet, et afin d'achever un niveau de sécurité aussi élevé que possible toute en préservant la facilité d'utilisation et d'accès à l'information par la personne concernée (patient ou soignant), nous allons utiliser la technologie blockchain avec la technologie cloud IPFS(InterPlanetary File System) pour le stockage des données et avec une méthode de contrôle basée sur les attributs et le chiffrement CP-ABE. Ces trois technologies nous permet de de garantir la confidentialité, l'intégrité, et la traçabilité des données médicales dans le système e-santé.

Ce présent mémoire est composé de quatre chapitres :

- Le premier chapitre consiste à donner un aperçu sur les applications e-santé et les technologies blockchain et cloud computing.
- Dans le deuxième chapitre, nous présentons la cryptographie et les modèles de contrôle d'accès ainsi que les algorithmes de chiffrement par attribut ABE.
- Dans le troisième chapitre, nous expliquons notre solution.
- Dans le quatrième chapitre, nous décrivons les étapes de réalisation et de mise en place de notre plateforme.

Enfin, nous clôturons notre mémoire par une conclusion et quelques perspectives

CHAPITRE 1 : Les applications e-santé et Blockchain.

Dans ce chapitre, nous allons présenter les systèmes E-santé. Ensuite, nous décrivons la nouvelle technologie blockchain qui est au centre de plusieurs recherches dans différents domaines, et la possibilité d'utiliser cette technologie décentralisée pour renforcer la sécurité. Enfin, nous passons en revue le cloud computing qui répond au besoin de stockage des données volumineux, telle que le cas dans les applications e-santé

1. Le Système E-santé

L'e-santé est un domaine clinique d'information concentré où de nombreuses informations sont créées, utilisées et partagées régulièrement. Le stockage et le partage de cette énorme base d'information est également très importants et difficiles en raison de la sensibilité de l'information et de facteurs restrictifs comme la sécurité et la confidentialité de l'information. Dans ce domaine, le partage et la gestion sécurisés de l'information sont essentiels car le personnel médical continue de partager les informations médicales du patient avec les autorités concernées pour des mises à jour et un suivi régulier. Dans cette section, nous allons définir le terme d'E-santé, présenter des dossiers médicaux électroniques et personnels et citer les exigences de sécurité dans ce type de système.

1.1 Définition de l'e-santé :

L'E-santé peut d'abord être définie comme l'utilisation des Technologies de l'Information et de la Communication (TIC) pour le transfert et l'échange à distance de données en matière de santé, que ce soit à des fins d'information, de formation, de diagnostic, de traitement, de recherche ou de gestion. L'OMS (Organisation Mondiale de la Santé) a néanmoins rappelé récemment que la « santé » ne se limite pas à la seule dimension des maladies, qu'elles soient aiguës ou chroniques, mais que la santé correspond à un « état de complet de bien-être physique, mental et social ». Le terme « santé » intéresse également les thèmes liés aux limitations d'activités et de restriction de participation à la vie en société. En fait, l'E-santé concerne deux domaines complémentaires [1]:

1. L'ensemble des systèmes d'information du domaine de la santé (et pas seulement de la médecine, le champ d'application étant ainsi très large) incluant les méthodes et technologies d'exploitation et d'analyse des données collectées à partir de ces systèmes d'information variés et diversifiés ;

2. La télésanté qui comporte deux volets :

- La télémédecine qui recouvre toutes les techniques et applications permettant d'intervenir à distance pour établir des diagnostics, mettre en œuvre des thérapeutiques, surveiller des traitements, assurer et suivre des soins coordonnés ;
- Les téléservices pour la vie courante et le bien-être social, qui permettent de mettre en œuvre des solutions d'aide et de vigilance vis-à-vis de personnes fragiles et de

compensation de la perte d'autonomie. Ces téléservices prolongent le « soin » médical par le « prendre soin » social (en dehors du cadre purement sanitaire).

1.2 Dossier Médical électronique :

Le Dossier Médical Electronique (DME) est un dossier électronique d'information sur la santé des patients généré par une ou plusieurs consultations dans un contexte de prestation de soins.

Ces renseignements comprennent des données démographiques sur les patients, des notes d'étape, des problèmes, des médicaments, des signes vitaux, des antécédents médicaux, des immunisations, des données de laboratoire et des rapports de radiologie. Le DME est conçu pour automatiser et rationaliser le flux de travail de l'hôpital, la clinique… [2]. Un DME pleinement fonctionnel devrait comprendre :

1) Le dossier des patients.
2) Le système de communication des ordonnances ou la saisie informatisée des ordonnances des médecins.
3) Le système de soutien à la prise de décisions.
4) Gestion des documents et des images.
5) Portail patient.
6) Gestion des documents et des notes internes et externes.
7) Statistiques et rapports.

Les données du DME sont généralement recueillies par les fournisseurs de soins de santé du patient employés par un organisme de maintenance sanitaire qui possède et exploite le système [1]. Ce système est conçu pour fournir une image complète de l'état du patient en tout temps. Ceci est particulièrement bénéfique à mesure que le système de santé devient de plus en plus complexe, puisque les domaines d'expertise en médecine se sont rétrécis et que plus de spécialistes sont impliqués dans le processus thérapeutique. Cela souligne l'importance d'avoir des dossiers médicaux informatisés accessibles à un éventail de professionnels de la santé pour améliorer les soins aux patients. Les DME peuvent également améliorer la productivité des médecins puisqu'ils peuvent accéder aux renseignements médicaux avant que le patient et le médecin ne se rencontrent et éviter les malentendus causés par l'écriture manuscrite [3].

Bien que ce système présente de nombreux avantages, les taux d'adoption sont relativement faibles dans les cliniques communautaires [4] qui font face à un certain nombre d'obstacles [5]. Un facteur majeur est le coût financier du système [6]. Pour surmonter cet obstacle, le gouvernement américain propose des incitations financières [7]. Par conséquent, les taux de mise en œuvre sont passés de 18 % en 2001 à 78 % en 2013 aux États-Unis [8]. La communauté des soins de santé a lentement accepté l'e-santé comme un outil qui peut aider le personnel médical.

Les statistiques de ce domaine on Algérie sont presque inexistante ce qui montre un taux d'adoption tré faible.

Le DME a aidé les médecins à identifier les maladies en fonction des symptômes et des caractérisations des patients découverts dans les études déclarées. Les hôpitaux ont pu retracer les patients qui répondent aux critères en faisant simplement une recherche dans la base de données des DME. Par exemple, Lin et al. [9] ont montré qu'en examinant les caractéristiques structurées et non structurées du système chez les patients souffrant de polyarthrite rhumatoïde, le DME a réussi à déterminer lesquels d'entre eux souffraient également d'une toxicité hépatique induite par le méthotrexate. Les hôpitaux peuvent également effectuer des recherches dans le système pour repérer les patients ayant déjà reçu un diagnostic afin de surveiller leur consommation de médicaments (en faisant des renvois entre les prescriptions et les fonctions de facturation), [10] l'apparition de la maladie à l'étude et les modes de soins. [11]

1.3 Dossier médical personnel :

Les Dossiers Médicaux Personnels (DMP) sont des dossiers médicaux électroniques contenant des données médicales et des renseignements sur un patient qui sont tenus à jour par les patients eux-mêmes. Les patients peuvent accéder aux DMP en ligne et consulter les résultats des tests, les ordonnances, les allergies, etc. Ces dossiers médicaux peuvent être gérés par les personnes en ajoutant des antécédents médicaux, des renseignements personnels ou simplement pour surveiller leur santé. Les DMP sont un élément vital de l'intervention de transition des soins où les patients plus âgés sont encouragés à participer plus activement à leur processus thérapeutique [12]. Dans une interaction patient-médecin, les DMP peuvent fournir au patient un langage partagé avec le médecin. Les patients mieux informés peuvent constater que les visites à l'hôpital ou les rendez-vous chez le médecin sont plus positifs. Une autre caractéristique des DMP et de la politique de soins axés sur le patient est que les patients peuvent maximiser leurs bienfaits pour la santé en organisant l'information de façon logique. Le DMP est conçu pour alléger le fardeau de la maladie (en donnant accès à l'information) et est utile pour maintenir le bien-être [13].

Les DMP ont été discutés pour la première fois à la fin des années 1970 [14] ; cependant, la plupart des études ont été menées et publiées au début des années 2000 depuis que les DMP sont devenus plus répandus dans le secteur des soins de santé à l'époque. L'idée originale découlait de la nécessité d'individualiser les technologies et de rendre les dossiers médicaux plus accessibles au public. Certains DMP offrent des services à valeur ajoutée comme la prise de rendez-vous (avec le même médecin ou un médecin différent), les interactions avec les médicaments, les rappels sur ordonnance, les rendez-vous chez le médecin, etc.

Deux des efforts du DMP les plus notables sont Google Health, qui a été arrêté le 1er janvier 2012, et Microsoft HealthVault, qui est toujours disponible. Comme l'a clairement indiqué la décision de Google d'arrêter Google Santé, son adoption par les individus a été plus lente que prévu. Malheureusement, aucune statistique n'est apparemment dans le domaine public sur le nombre d'utilisateurs de HealthVault (actuellement disponible uniquement aux États-Unis et au Royaume-Uni).

L'utilisation des systèmes électroniques dans le domaine des soins de santé est en hausse pour tous les groupes d'âge en Europe [15], un aperçu complet de l'utilisation des DMP [16] a montré que les principales raisons de l'adoption des DMP étaient la communication patient-médecin, et le mode de vie et l'autogestion de la santé. Ainsi, les personnes atteintes de maladies chroniques, ou celles qui s'occupent de personnes âgées ont exprimé des attitudes plus favorables à l'utilisation de DMP, bien que cela n'implique pas nécessairement une utilisation réelle. Outre le taux relativement faible d'adoption des DMP, la conclusion concernant les avantages de l'utilisation des DMP n'étaient pas concluantes en ce qui concerne les résultats pour la santé [17], à savoir qu'il n'y avait aucune preuve indiquant une amélioration de l'état de santé des utilisateurs. De toute évidence, la principale préoccupation exprimée par les utilisateurs potentiels de DMP était les atteintes à la sécurité et à la confidentialité, comme on pouvait s'y attendre, mais les préoccupations relatives à la convivialité étaient également évidentes, en particulier pour les personnes ayant des déficiences cognitives ou une faible maîtrise de l'informatique [18]. D'un autre côté, les médecins craignaient d'avantage l'exactitude de l'information sur la santé contrôlée et gérée par les patients. Le manque d'intérêt des médecins a également été considéré comme un obstacle important puisque l'accès à des renseignements médicaux complets et en temps opportun est essentiel à une utilisation satisfaisante et bénéfique des DMP.

Dans notre travail, nous allons utiliser ce genre de dossier avec quelques modifications notamment l'impossibilité d'écriture sur le dossier par le patient.

1.4 Exigences de sécurité des applications e-santé :

Les exigences de sécurité pour les applications e-santé sont difficiles, tels que l'authentification mutuelle, l'utilisateur anonymat, non-traçabilité, perfect-forward-secret, accord de clé de session, et la résistance aux attaques pour assurer la confidentialité et la sécurité des données [1].

- **Authentification mutuelle :**

Cela peut être réalisé avec l'utilisation de protocoles d'authentification tels qu'Authentification Kerberos. Nous avons analysé à partir de la littérature que le transport la sécurité de la couche (TLS) / la couche de socket sécurisée (SSL) assure le flux de communication mais ne peut pas vérifier ou le dispositif de communication, qui peut être vérifiée par authentification mutuelle. Il permet seul l'utilisateur autorisé à accéder aux informations du serveur.[1]

- **Anonymat :**

Si un attaquant obtient l'identité de l'utilisateur, la confidentialité du patient peut être compromise, Par conséquent, l'anonymat est l'une des exigences de sécurité dans l'e-santé. Le patient et L'identité du médecin doit être prouvée lors de la phase de demande de connexion. Cependant, il est difficile d'obtenir l'identité des patients et des médecins car ils sont cryptés.[1]

- **Non traçabilité :**

Si un attaquant retrace les exercices communication de clients spécifiques, alors il / elle peut deviner l'identité réelle des patients avec une probabilité plus élevée.

Cela entraîne une violation de la vie privée des utilisateurs. Un attaquant ne peut pas décider des exercices de communication d'un utilisateur spécifique.[1]

- **Secret de transmission parfait (pfs) :**

Perfect Forward Secrecy (PFS) est utilisé pour l'accord de clé, qui protège les sessions précédentes contre l'accord futur des mots de passe ou des clés privées en créant une clé de session pour chaque session. Ici, un attaquant ne peut pas accéder aux clés de session, qui ont été créées dans le passé sessions ; même si n'importe qui peut accéder à la clé privée de l'utilisateur, il ne peut pas affecter car la clé de session est chiffrée avec plusieurs algorithmes.

2 Technologie blockchain.

Blockchain a gagné en popularité car il est une base de données distribuée et décentralisée. Initialement, Blockchain a été introduit par Satoshi Nakamoto un pseudonyme utilisé par la ou les personnes ayant développé la cryptomonnaie Bitcoin, Il a été utilisé pour l'échange d'argent électronique entre le réseau distribué de divers clients sans inclure un tiers. Mais plus tard, il a été mis en œuvre dans des domaines non financiers étant considéré comme une technologie générale. Avec la demande croissante de la technologie blockchain dans de nombreux domaines, les domaines de la santé ont également identifié certains de ses cas d'utilisation comme des applications blockchain comme nous allons expliquer dans cette section. Mais avons cela, nous allons présenter les notions de bases de blockchain. [19]

2.1 Définition du Blockchain

Blockchain est une chaîne de blocs qui sont connectés ensemble et sont en croissance continue en stockant les transactions sur les blocs (Figure 1) :

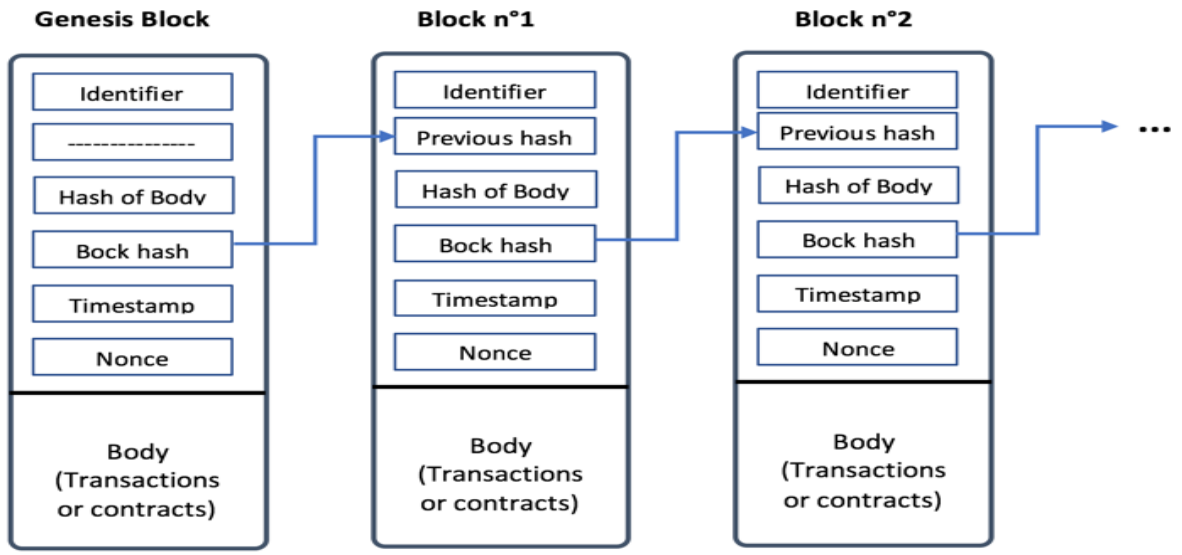

Figure 1 : Structure d'un Blockchain [20]

Dans ce qui suit, nous définissons les concepts de base de blockchain que nous utilisons tout au long du document :

- Les **blocs** sont des lots de transactions avec un hash du bloc précédent dans la chaîne. Ceci relie les blocs ensemble (dans une chaîne) parce que les hachures sont cryptographiquement dérivées des données du bloc. Cela empêche la fraude, car un changement dans n'importe quel bloc de l'histoire invaliderait tous les blocs suivants que toutes les haches suivantes changeraient et tout le monde exécutant le blockchain remarquerait.
- Une **transaction** est toute opération qui consiste à modifier l'état du blockchain en ajoutant des données qui seront stockées de façon irréversible. Elle peut être sous forme d'échange de jeton ou de monnaie entre les utilisateurs
- Un **Contrat intelligent** « Smart contract » présente l'exécution d'un code stocké dans le blockchaine.
- Un **nœud** est un ordinateur qui est connecté à un réseau blockchain. Le nœud ou l'ordinateur prend en charge le réseau. Il le soutient par la validation et le relais des transactions. En même temps, il obtient également une copie complète du blockchain.
- Les **mineurs/validateurs** sont des utilisateurs du blockchain dont le rôle est de valider les transactions qui circulent dans ce dernier.
- Le **Mining** est le processus de validation d'un bloc de transactions à ajouter à la blockchain qui est réalisé par le mineur.
- Un **Nonce** : une abréviation de "numéro utilisé une seule fois," qui est un nombre ajouté à un bloc haché ou crypté dans un blockchain, lorsqu'il est haché de nouveau afin de répondre aux restrictions de niveau de difficulté.
- Le **hash d'un bloc précédent (Previous Hash)** permet d'assurer que le bloc n'a pas été altéré par un tiers. En effet, changer n'importe quelle variable d'un des hashes dans un bloc donné provoquerait un effet domino, modifiant tous les haches dans le blockchain.
- **Bloc Genesis** : est le premier bloc de transaction dans le blockchain.

2.2. Architecture

La figure 2 illustre l'architecture de blockchain en expliquant l'ensemble du processus d'une transaction envoyée par un utilisateur sur le réseau blockchain :

1. Une nouvelle transaction envoyée par un utilisateur sur le réseau blockchain suggère soit la création d'un nouveau bloc si le dernier bloc est plein (car un bloc peut contenir plusieurs transaction, le nombre de ces dernières dépend de la taille de leurs contenu), sinon l'ajout de cette transaction au dernier bloc existant. Les blocs dans le blockchain sont utilisés pour garder les transactions en eux et ces blocs sont distribués sur tous les nœuds connectés dans le réseau. Cette transaction placée à l'intérieur d'un bloc est diffusée à tous les nœuds du réseau. Tous les nœuds du réseau ont une copie du blockchain complète qui les aide dans le processus de vérification. Lorsqu'un bloc contenant la transaction de l'utilisateur est diffusé sur tous les nœuds connectés, ils vérifient que le bloc n'est pas altéré par quelque moyen que ce soit. Si cette vérification aboutit à un succès, les nœuds ajoutent ce bloc dans leur propre copie de blockchain.

2. Ce processus entier quand le bloc étant ajouté sur le blockchain est fait par les nœuds atteignant un consensus où ils décident quels blocs sont valides pour être ajoutés sur le blockchain et qui ne le sont pas. Cette validation est effectuée par les nœuds connectés à l'aide d'algorithmes de consensus pour vérifier la transaction et s'assurer que l'expéditeur est un parti authentifié du réseau. Lorsqu'un nœud réussit à effectuer la validation, ce nœud est récompensé par une crypto-monnaie. Ce processus de validation de la transaction est connu sous le nom de mining et le nœud effectuant cette validation est connu sous le nom de mineur.
3. Une fois la validation terminée, le bloc est ajouté au blockchain.
4. Une fois le processus de validation terminé, la transaction est terminée.

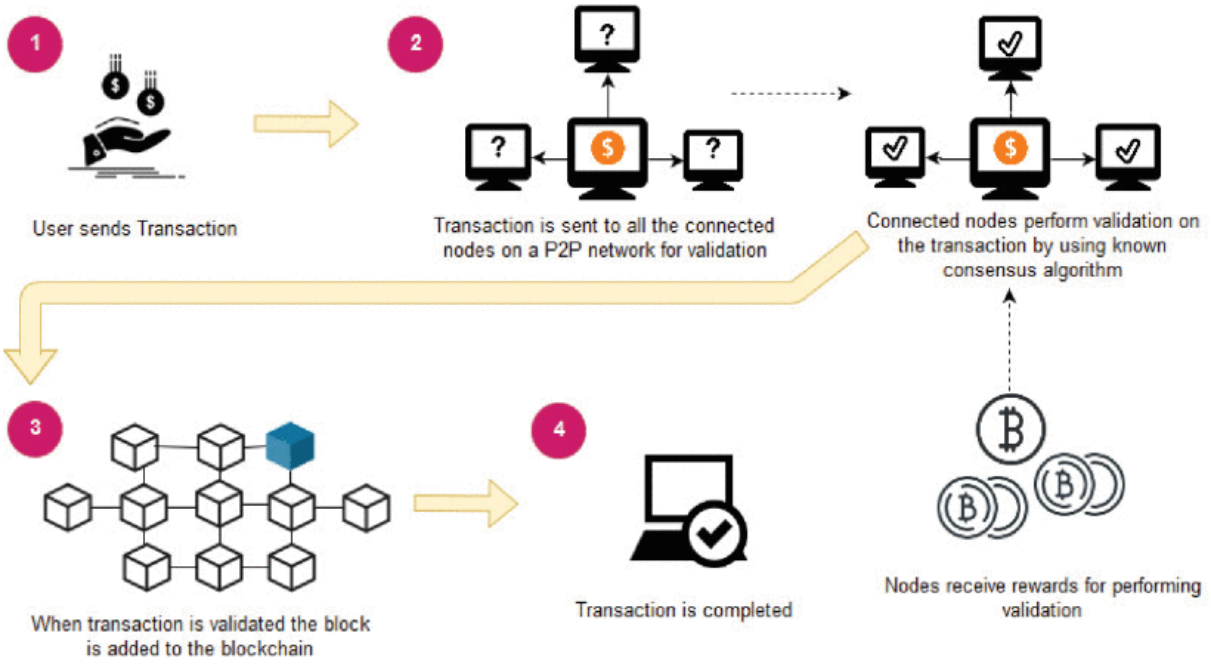

Figure 2: Architecture du Blockchain[21]

2.3. Les algorithmes Consensus

Dans les systèmes distribuée ou décentralisés, tout le monde (tous les nœuds du réseau Blockchain) doivent être d'accord sur l'ordre des transactions. Les mineurs y contribuent en résolvant des énigmes difficiles sur le plan informatique afin de produire des blocs, qui servent à protéger le réseau contre les attaques.

Toutefois il n'y a pas de protocole de consensus parfait. Le protocole de consensus doit faire un compromis entre la cohérence, la disponibilité et la tolérance aux pannes de cloison [22]. Il doit également aborder le problème des généraux byzantins qui dit qu'il y aura certains nœuds malveillants qui essai délibérément de compromettre le processus de consensus. Nous allons donner une description de deux protocoles de consensus blockchain populaires qui peuvent traiter efficacement le problème des généraux byzantins.

2.3.1. Algorithme de Proof of work (Pow):

Le protocole de proof of work, connu sous le nom d'Ethash, exige des mineurs de passer par une course intense d'essai et d'erreur pour trouver le nonce pour un bloc. Seuls les blocs avec un nonce valide peuvent être ajoutés à la chaîne.

Lors de la course pour créer un bloc, un mineur mettra à plusieurs reprises un ensemble de données qui ne peuvent pas être obtenues en téléchargeant et en exécutant la chaîne complète, à travers une fonction mathématique. Ceci est pour générer un mixHash qui est en dessous d'un nonce cible. La meilleure façon de le faire est par essais et erreurs.

La difficulté détermine la cible pour le hachage. Plus la cible est basse, plus l'ensemble des hachages valides est petit. Une fois généré, cela est incroyablement facile à vérifier pour les autres mineurs et clients. Même si une transaction devait changer, le hachage serait complètement différent, signalant la fraude.

Le hachage rend la fraude facile à repérer. Mais PoW comme un processus est également un grand dissuasif à attaquer la chaîne.

2.3.2. Algorithme de Proof of stake (Pos) :

Cet algorithme oblige les utilisateurs à miser sur leur monnaie pour devenir un mineur dans le réseau. Les mineurs sont responsables de la même chose que les mineurs en proof of work: commander des transactions et créer de nouveaux blocs pour que tous les nœuds puissent s'entendre sur l'état du réseau.

La proof of stake(pos) comprend un certain nombre d'améliorations au système de proof of work:

- Meilleure efficacité énergétique – on n'a pas besoin d'utiliser beaucoup de blocs d'extraction d'énergie
- Réduire les obstacles à l'entrée, réduire les exigences en matière de matériel – on n'a pas besoin de matériel d'élite pour avoir une chance de créer de nouveaux blocs
- Une plus grande immunité à la centralisation – la proof of stake devrait mener à un plus grand nombre de nœuds dans le réseau
- Un soutien plus fort pour les chaînes– une mise à niveau clé dans la mise à l'échelle du réseau.

Dans notre travail, nous allons utiliser le proof of work car la méthode proof of stake est encore dans l'étape expérimentale de son développement ; il n'y a pas une version stable et utilisable à cette date.

2.4 Les caractéristiques principales de blockchain :

Blockchain est une nouvelle technologie avec plusieurs caractéristiques qu'on ne trouve que dans cette dernière et la plupart de ces caractéristiques ont un lien direct avec nos objectifs de sécurité visés dans notre projet :

- **Désintermédiation :**

La technologie blockchain permet d´échanger sans le contrôle d'un tiers. La validation et l'ajout d'un bloc résultent d'un consensus entre les utilisateurs-mineurs, qui repose sur la possibilité de vérifier leur travail de validation et qui rend inutile le contrôle par une institution de référence [23]. Tout est effectué sans l'intervention d'une autorité centrale, les utilisateurs opèrent la surveillance, et se contrôlent mutuellement, assurant la certification des sauvegardes et leurs cohérences [24].

- **Transparence :**

Une fois qu'un document est inscrit sur le blockchain, cela suffit à prouver que ce dernier existe bien à l'instant T et qu'il n'a pas été modifié. Le blockchain est qualifiée de transparent car tout le monde peut le télécharger dans son intégralité et vérifier à tout moment son honnêteté [25]. Tous les utilisateurs du blockchain peuvent ainsi voir les transactions présentes et passées [26]

- **Persistance :**

Les transactions enregistrées dans un nœud Blockchain sont considérées comme persistantes car elles se propagent à travers le réseau, où chaque nœud maintient et contrôle ses enregistrements. Tant que la majorité des nœuds sont bénins, la persistance est maintenue. Plusieurs propriétés sont dérivées de cette caractéristique, y compris la transparence et l'immutabilité (résistance au tempérament). Cette transparence et cette immutabilité signifient que les Blockchains sont vérifiables [27].

- **Validité :**

Contrairement à certains systèmes distribués, les blockchains ne nécessitent pas l'exécution de chaque nœud. Les transactions, ou blocs, diffusés dans un système Blockchain seraient validés par d'autres nœuds. Ainsi, toute falsification pourrait être détectée facilement. Ce système comprend trois rôles principaux : (1) les proposants qui proposent une valeur, (2) les accepteurs qui valident et décident de la valeur à prendre, et (2) les apprenants qui acceptent la valeur choisie [28].

- **Anonymat et Identité :**

L'anonymat est la principale caractéristique des blockchains publiques. L'identité dans ce système peut être déliée à une identité de l'utilisateur dans le monde réel. Un utilisateur peut obtenir de multiples identités pour éviter d'être exposé [29]. Aucune entité centrale n'a besoin de conserver des renseignements personnels. En conséquence, selon les informations de la transaction, l'identité du monde réel ne peut pas être obtenue, préservant une certaine quantité de vie privée. D'autre part, l'identité est généralement requise dans les systèmes qui sont exploités et gouvernés par des entités connues dans les paramètres tels que les blockchains privées et autorisées.

- **Audibilité :**

Enregistrer l'horodatage et les informations persistantes permettent de vérifier et de tracer facilement les enregistrements précédents à travers des nœuds dans un réseau Blockchain. Le degré d'auditabilité dépend des types de systèmes Blockchain et de leurs implémentations. Les blockchains privées sont les moins vérifiables car les nœuds sont administrés par une entité, les blockchains autorisées viennent en second lieu dans lesquelles certains accords, tels que les données cryptées, peuvent empêcher que les informations soient entièrement vérifiables, et les blockchains publiques sont les plus élevées car les nœuds sont vraiment décentralisés.

- **Fermeture et ouverture (Closedness and Openness):**

Les blockchains ouverts s'appuient sur des nœuds publics pour tenir des registres de transactions. Par conséquent, n'importe qui peut publier une transaction et rejoindre le système en suivant un ensemble de règles et d'informations à l'intérieur de ce Blockchain qui est public. Les blockchains autorisées sont considérées comme semi-ouvertes car les nœuds sont pré-spécifiés ou validés avant l'assemblage. Ils se situent entre les blockchains publiques et privées. L'information à l'intérieur de ce Blockchain est contrôlée par les politiques du consortium, qui peut réglementer l'information à être entièrement ouvert, partiellement ouvert, ou fermé. Tout comme les blockchains autorisées, les blockchains privées contrôlent la sélection des nœuds et le degré d'ouverture des données en fonction des politiques. Cependant, ils comptent sur une seule entité ou un seul propriétaire.

2.5 Différences entre Blockchain et Bases De Données.

Le blockchain peut être considérée comme une base de données. Cependant, ces deux entités sont fondamentalement différentes. Une base de données utilise une architecture réseau client-serveur, alors qu'un blockchain est purement basée sur une architecture réseau peer-to-peer (figure 3). Le tableau 1 illustre certaines des différences critiques entre un blockchain et un système de base de données.

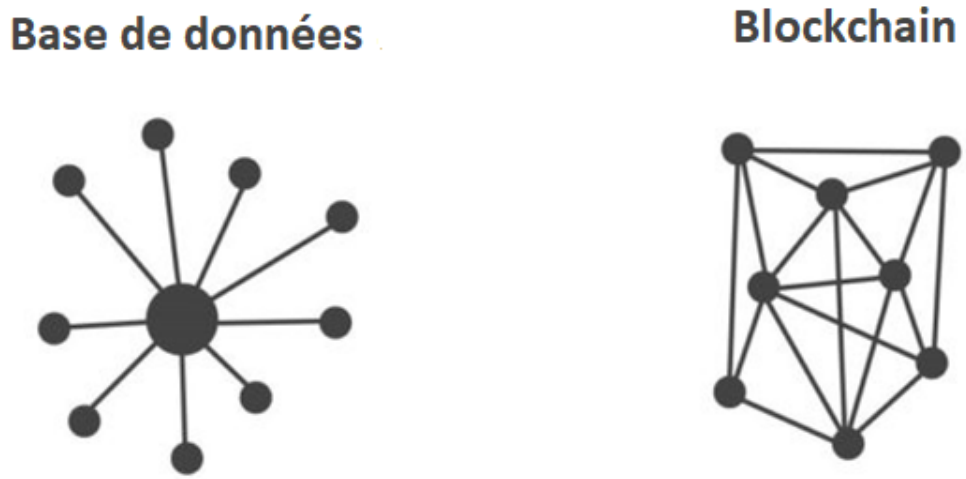

Figure 3: architecture d'une base de données et d'un blockchain

Propriétés	Blockchain	base de données
Opérations	(+)L'opération d'insertion et de lecture seulement	(-)peut effectuer des opérations (création , lecture , modification , suppression)
Réplication	(+)Réplication complète du bloc sur chaque nœud	• (-)maître-esclave • (-)multi-maître
Consensus	(+)La majorité des nœuds s'entendent sur le résultat des transactions	(-)Transactions distribuées
Invariants	(-)Tout le monde peut valider les transactions sur le réseau	(+)Contraintes d'intégrité
construction de la confiance	(+) Peut fonctionner sans aucune partie de confiance	(-) Besoin d'une partie centrale de confiance
confidentialité des données	(+) (Par défaut) Tous les nœuds ont accès (lecture) sur les données	(-) Elle restreint l'accès aux personnes autorisée
Robustesse/tolérance aux pannes	(+)Les données sont réparties entre les nœuds	(-) Les données sont stockées dans la base de données centrale
Performance	(-) Prendre un peu de temps pour arriver à un consensus dans le réseau de (1 jusqu'à 2 minutes)	(+) Exécution/mise à jour immédiate
Redondance	(+) (Par défaut) Chaque nœud participant a la dernière copie du ledger	(-) Seule la partie centrale a une copie
Sécurité	(+) (Par défaut) Utiliser des mesures cryptographiques	(-) Utilise le contrôle d'accès traditionnel

Tableau 1 : Comparaison entre blockchain et base de données [30]

Dans ce qui suit, nous détaillons les points de différence entre un blockchain et une base de données :

- ## Contrôle décentralisé ou centralisé

Une blockchain est un système décentralisé et trustless. Cela signifie que si deux parties ne se font pas confiance et souhaitent partager ou échanger des informations sensibles sans l'intervention d'un tier, cela est possible avec la technologie blockchain. Dans blockchain, les transactions sont généralement traitées par les nœuds du réseau qui agissent en tant qu'intermédiaires, garantissant que chaque utilisateur crée le même système d'enregistrement partagé au même moment. L'avantage principal de la technologie blockchain est qu'elle élimine la nécessité d'un système de contrôle centralisé. Par ailleurs, une base de données est un système

centralisé, ce qui signifie qu'on doit faire confiance à un tier pour la gestion. C'est ce tier qui donne le droit de lire et d'écrire les données stockées dans la base de données. Et comme tout cela est centralisé, la maintenance des bases de données est assez facile et le résultat est élevé.

- ## Performance

Les blockchains sont des plates-formes idéales pour les transactions, elles sont considérées comme lentes en comparant aux transactions numériques telles que Visa et PayPal [30]. Les recherches continuent constamment afin d'améliorer la technologie blockchain, la nature de la technologie blockchain requière le sacrifice de la vitesse. Lors d'une transaction les nœuds traitant la transaction compare les résultats avec d'autre nœuds avant de finaliser la transaction, ce qui signifie que le temps requis pour une transaction avec la technologie et plus long que d'autre. D'autre part, les bases de données, existe depuis longtemps et sont devenues plus performantes avec le temps.

- ## Confidentialité

Bitcoin est devenu populaire grâce à son écriture et lecture non contrôlée, ce qui veut dire que toute personne peut écrire et lire sur ce blockchain. À l'opposé, une base de données centralisée est contrôlée en écriture et en lecture par un tiers. Elle peut être configurée de façon à ce qu'elle ne permette qu'à des personnes spécifiques d'écrire dans la base de données ou de lire la base de données.

Les informations stockées dans des blockchains ne peuvent pas être modifiées autrement dit ils sont immutables, on peut qu'ajouter des blocs sur le blockchain.

Pour modifier une information stockée dans le blockchain, un consensus à l'échelle du réseau est requis. Autrement dit, chaque nœud sur le réseau doit approuver le changement. Comme il existe généralement des centaines, des milliers de nœuds sur un réseau blockchain, il serait pratiquement impossible de faire un tel changement. Cependant dans une base de données, on nécessite la permission des administrateurs ; ou d'un logiciel particulier pour éditer ou supprimer des informations.

Une base de données ne contient souvent que les informations les plus récentes ou les informations que l'administrateur juge utile ou nécessaire, un blockchain contient toutes les informations qui y ont été ajoutées.

Par conséquent, un blockchain peut être adaptée pour des éléments d'information qui nécessitent un historique , tels que des certificats de mariage, de naissance et de décès et d'autres types de documents juridiques. Une base de données est généralement mieux adaptée pour les informations sujettes à changement ; comme des dossiers de recherche qui parfois nécessite une mise à jour pour maintenir l'exactitude de l'information et éviter la confusion.

- **Impossibilité de consensus distribué dans les bases de données**

Le problème de parvenir à un accord entre les processus à distance est l'un des problèmes fondamentaux de l'informatique distribuée qui est au cœur de nombreux algorithmes pour le traitement des données distribuées, la gestion de fichiers distribués et la tolérance aux erreurs d'applications distribuées.[31]

La solution la plus adapté pour résoudre ce genre de problème est d'avoir une copie de toutes les données sur chaque membre du réseau qui est la définition même de la nouvelle technologie blockchain une solution pour les données distribuées par consensus.

2.6 Types de Blockchain

Il existe deux types des réseaux de blockchain : publics et privés. Ces réseaux sont décentralisés et partagés entre leurs utilisateurs pour enregistrer toutes les transactions peer-to-peer sans qu'une tierce partie ait habituellement de confiance pour les autoriser [32]. Cependant, certaines caractéristiques distinctes rendent les blockchains publiques et privées différents.

Les blockchains privées ont un taux de traitement des transactions très élevé avec très peu de participants autorisés. Par conséquent, il faut moins de temps pour obtenir le consensus pour le réseau et plus de transactions peuvent être traitées en une seconde. Par opposition, les blockchains publiques ont un taux de traitement des transactions très limité. Les mécanismes de consensus tels que Bitcoin proof of work (PoW) dans les blockchains publiques ont besoin de l'ensemble du réseau pour atteindre le consensus sur l'état des transactions. Les blockchains publiques ont également la confidentialité de l'information risquée en raison de leur nature inhérente. Les blockchains publiques s'appuient sur un processus d'ajout de données seulement menant à un stockage de données immutable [33]. De plus, dans les blockchains publiques, le nœud entier doit s'entendre sur tout changement puisqu'il enregistre les mêmes informations. Ainsi, tout changement doit être enregistré dans tous les blocs suivants et il faut plus de temps pour extraire un seul bloc du blockchain [34]. De plus, pour assurer l'intégrité du blockchain, tous les blocs sont liés au bloc genesis [35]. Les blockchains privées ont une très forte confidentialité des données où tout changement peut être fait simplement lorsque tous les nœuds conviennent que les données peuvent être modifiées par consensus [36]. Un autre problème avec les blockchains publiques est le contrôle des utilisateurs dans le téléchargement d'informations. Par exemple, si quelqu'un dans le système télécharge des informations sensibles dans le système, il n'y a aucun moyen de changer cette action [37].

Bien que le blockchain publique dispose d'un nombre illimité de nœuds anonymes, chaque acteur peut communiquer de manière laïque sur la base de la cryptographie [38]. Chaque nœud a une paire de clés privées / publiques. Par conséquent, un blockchain publique ne nécessite pas de faire confiance à quiconque utilise le réseau. Tous les membres du réseau sont incités à agir conformément au contrat pour obtenir les meilleurs résultats du réseau. Ainsi, la validation et la vérification d'une transaction peut être effectuées sans qu'une partie de confiance soit nécessaire. Cela conduit à une autre force principale des blockchains publiques, c'est-à-dire qu'elles sont très transparentes. Chaque transaction dans un blockchain publique est ouverte au

public pour vérification. Cependant, dans les blockchains privées, seules les parties de confiance peuvent être présentées sur le réseau pour vérifier et valider les transactions. Les blockchains publiques sont plus décentralisées avec de nombreux nœuds et il est donc plus difficile pour tout mauvais acteur d'apporter des changements au réseau. Cependant, les blockchains privées ont moins de nœuds de sorte qu'il est plus facile d'obtenir le contrôle sur le réseau par tout mauvais acteur. Par conséquent, le risque de piratage et de manipulation de données est plus élevé dans les blockchains privées par rapport aux blockchains publiques. Ainsi, on peut affirmer que les blockchains publiques sont plus sécurisées. De plus, les blockchains publiques n'exigent aucun coût d'infrastructure pour établir le réseau [39].

Dans le tableau suivant, nous trouvons une comparaison entre les blockchains privés et publiques selon les mesures d'évaluation: Coût par transaction, Rendement, Confiance du système, Évolutivité et Maintenance [29].

Type	Cout par Transaction	Performance		confiance du Système	Évolutivité	Maintenance	Openness
		Débit	Latence				
Publique	Élevé	Faible	Élevé	Élevé	Élevé	Faible	Élevé
Privé	Faible	Élevé	Faible	Faible	Faible	Moyen	Moyen

Tableau 2: Comparaison Blockchain publique et privé[29]

Dans notre projet, nous allons utiliser un blockchain privé pour assurer la confidentialité des données de nos patients dans les applications e-santé. Cette solution est déjà utilisée par d'autres chercheurs comme nous le verrons dans la section suivante.

2.7 Application e-santé basé sur blockchain

Un blockchain a certains avantages tels que la sécurité, l'anonymat et l'intégrité des données sans intervention de tiers. Ces avantages en font un choix raisonnable pour stocker les dossiers médicaux des patients, car l'innovation technologique dans le secteur de la santé a fait de la sécurité des données médicales une priorité absolue. Un certain nombre de chercheurs ont également identifié que l'utilisation de la technologie blockchain dans les soins de santé serait une solution réalisable [21]. Dans ce qui suit, nous présentons les principaux problèmes rencontrés dans ces applications et les solutions proposées pour y remédier.

2.7.1. Problèmes rencontrés

Lors de l'utilisation de blockchain dans E-santé, il est possible de rencontrer des problèmes suivants :

- **Confidentialité et sécurité de l'information**

En ce qui concerne la gestion ou le stockage des données, les préoccupations relatives à la confidentialité et à la sécurité des renseignements sont toujours présentes. Comme mentionné ci-dessus, le besoin de tiers pour l'achèvement de la transaction dans le blockchain a été éliminé [40]. Le blockchain exige l'accord des participants pour vérifier le dossier au lieu de l'approbation d'un tiers, ce mécanisme entraîne un risque pour la sécurité et la confidentialité de l'information, car tous les participants peuvent avoir accès aux dossiers d'information et la vie privée peut être violée. En cas d'urgence, le patient peut permettre à un ou plusieurs membres du personnel autorisés d'avoir accès aux renseignements sur le patient et ces représentants peuvent donner accès aux renseignements sur le patient à un certain nombre d'autres membres du personnel, ce qui peut mettre les données en péril.

- **Stockage limité :**

Comme nous l'avons mentionné précédemment, le blockchain a été proposée pour l'enregistrement et le traitement de l'information sur les transactions, de sorte qu'elle ne nécessitait pas un stockage énorme. Maintenant qu'il a été introduit dans le domaine des soins de santé, son stockage limité est devenu un problème majeur en termes de stockage de données, car les données médicales (comme les antécédents médicaux du patient, les rapports de tests, les radiographies, le nombre de patients hospitalisés, …) nécessitent une grande capacité de stockage. En raison de l'augmentation de la base de données, il ralentit également la recherche de documents et l'accès à l'information qui ne convient pas à l'application où la vitesse est fortement requise.

- **Larges algorithmes :**

Blockchain utiliser de grands algorithmes pour les méthodes et techniques de sécurité qui peuvent conduire à des fuites d'informations. Les algorithmes peuvent être utilisés comme porte dérobée par les pirates pour briser la sécurité et entrer dans la base de données. C'est un enjeu majeur pour la protection des renseignements personnels des patients.

- **Interprétabilité :**

L'interprétabilité est également un problème majeur dans les applications e-santé basées sur le blockchain qui ne peut être évité. Comme mentionné précédemment, le partage de données a lieu lorsqu'il s'agit de partager les informations du patient avec des experts ou d'autres membres du personnel. De plus, les fournisseurs de blockchains et les fournisseurs de services communiquent entre eux, ce qui peut également présenter un risque pour le partage sécurisé des données [41].

- **Risque Cybersécurité :**

Les risques liés à la cybersécurité continuent d'augmenter dans les réseaux e-santé de façon directe et indirecte. Ces attaques comprennent le déni distribué de services, le vol de

renseignements personnels sur la santé et le rensomware. Les attaquants essaient de trouver des réseaux privés virtuels ou des dispositifs connectés à distance pour entrer dans les réseaux privés. S'il est trouvé, l'attaquant peut entrer dans le réseau pour voler les informations et les crypter avec des ransomwares. Les applications e-santé autorisés ne peuvent pas accéder aux informations ou aux dossiers médicaux des patients tant que la rançon n'est pas payée. Récemment, ammersmith Medicines Research a été frappé par rensomware à Londres. De plus, en mars 2020, le département de la Santé et des Services sociaux des États-Unis a signalé des voies de fait contre le DDos.

2.7.2. Solutions Proposées :

L'utilisation de blockchain dans les applications e-santé présente plusieurs problèmes comme mentionné précédemment, les solutions à ces problèmes se résument par les suivantes :

- **Utilisation de blockchain privé :**

La mise en œuvre du blockchain privée dans les industries de soins de santé est une meilleure option pour sécuriser les informations et aussi très bénéfique pour maintenir la confidentialité des données sensibles des patients. Le blockchain privée permet un accès limité à l'information et tous les participants n'ont pas accès aux données. Le blockchain privée utilise une gestion stricte d'accès aux données [42]. En outre, Dans cette technique de e-santé privés, seules les personnes ou les patients autorisés peuvent accéder et gérer leurs propres données et dossiers médicaux stockés dans le blockchain privée qui sera accessible uniquement aux personnes autorisées. Il peut être une solution efficace pour maintenir la confidentialité des données. Il a également éliminé la question des menaces à la vie privée et à la sécurité qui ont été abordées dans la section précédente [43].

- **Mécanisme de stockage cloud :**

Cette approche est bénéfique pour le stockage sécurisé des données, elles peuvent être partagées en toute sécurité dans le cadre du cloud. Cette approche donne également la liberté aux patients d'accéder et de contrôler leurs propres données. Comme nous l'avons vu dans la section précédente, le blockchain permet un stockage limité de l'information surtout dans les systèmes de soins de santé. Heureusement le stockage en nuage a éliminé le problème de stockage limité pour ce genre d'applications et profite du blockchain pour la gestion du stockage [13].

- **Preuve d'interopérabilité :**

Il s'agit d'un algorithme de consensus qui permettra aux applications e-santé d'exécuter les transactions de manière fluide et efficace sur la base de l'interopérabilité des participants au réseau. Cette approche fonctionnera sur une architecture à trois niveaux, telle que la plateforme Web, elle sera utilisée par le patient pour mettre à jour son accès pour la gestion des dossiers médicaux, deuxièmement, Cloud Middleware, Elle récupérera les informations du niveau précédent en utilisant les services REST. Le troisième niveau s'occupera de la gestion des nœuds dans le réseau blockchain. Diviser la fonction blockchain en plusieurs niveaux au lieu d'un seul niveau aide à interoper-ability [44].

- **Authentification Multifacteur :**

L'authentification multifactorielle peut être utilisée pour authentifier les participants. Comme il a été mentionné que pour apporter des changements à l'information ou la stocker en blocs ou en ajouter de nouveaux, il faut le consentement de 51 % de tous les participants [45]. Il est possible que des fraudeurs puissent voler l'entité du participant vérifié et apporter des changements à l'information. Pour éviter ce risque, l'authentification multifactorielle peut être utilisée pour authentifier les utilisateurs qui veulent accéder aux informations et les utiliser.

2.7.3. Principaux Travaux Existants

Theodouli et coll. [46], ont animé un échange de données sur les systèmes e-santé qui dépasse le système contemporain en libérant le pseudonyme de privatisation de l'identification des utilisateurs dont les données sont partagées et utilisées par le Centres de recherche. Le travail a également établi le concept d'un Algorithme de consensus, « preuve d'interopérabilité », qui permet aux institutions organisées sur le système en des transactions efficaces basées entièrement sur l'Interopérabilité des différents nœuds du Réseau. De plus, l'auteur a également suggéré une architecture à trois niveaux, la plateforme Web, qui sera utilisée par le patient pour le téléchargement des dossiers médicaux et la tenue d'un Access Management Suite, Cloud Middleware – sera utilisé pour maintenir les données extraites de la plateforme Web à l'aide de l'API REST et de faire appel aux Smart Contracts pour l'exécution des enregistrement spécifié de blocs plus récents et pour soutenir le consensus des nœuds à travers le réseau Blockchain qui est détenu dans le troisième niveau.

Wang et al. [47], on fait progresser un système e-santé parallèle développé autour de l'approche ACP développée sur un Blockchain réseau. Le système définit l'utilisation d'un Système parallèle de santé basé sur les connaissances thérapeutiques réelles et l'expérience des médecins et des patients, et l'utilisation d'un système d'intelligence artificielle qui détermine la mise en œuvre des Médecins Virtuels et des Patients Virtuels pour se soumettre et définir une approche Parallèle dans la dictée du traitement qui doit être effectué par les médecins sur les patients. Le deuxième segment de l'approche ACP pointe vers le segment même des expériences de calcul, où il combine la signification clinique et l'expérience des 4 pour déterminer la procédure clinique et expérimentale générique qui doit être effectuée sur les patients. Le troisième segment permet l'exécution parallèle entre la santé artificielle et le Système e-santé et le système de santé réel, qui régit à l'approche entre les médecins du monde réel et artificielle médecins définis par le logiciel. La proposition fondamentale de base du système parallèle repose sur le fait que les médecins artificiels ou définis par le logiciel effectueront les expériences et le calcul sur les patients en fonction des caractéristiques qui ont été alimentés au système, les résultats qui seront générés par le les médecins artificiels seront vérifiés et mis à jour par les médecins. Le système de santé complet a été en outre incorporé avec un réseau Blockchain ayant le consortium des médecins, hôpitaux, patients, bureaux de santé et les chercheurs médicaux, qui peuvent être utilisés pour l'examen et le partage des données.

Zainab Alhadhrami et al. [48], ont discuté des différents types d'architectures blockchains disponibles dans le présent scénario et ont discuté de la base de tous les types de blockchains et de la façon dont il peut être utilisé dans le secteur de la santé pour maintenir, valider et stocker

les données. En outre, le système qui est principalement sorti de la boîte pour stocker les données de soins de santé a été marqué pour être le consortium blockchain. Les blockchains de consortium sont essentiellement celles qui sont les blockchains à permission où le propriétaire de nœud, ainsi que les mineurs, obtenir le contrôle d'accès. En outre, le consortium blockchain travaille sur la théorie du consensus d'un nombre majoritaire d'intervenants ou les nœuds associés au réseau blockchain.

Liu et al. [49], ont proposé une architecture de blockchain avancée pour les systèmes e-santé. La principale préoccupation du travail a porté sur l'élaboration d'une solution de réseautage interopérable et adaptable pour le partage efficace et approprié des données sur les plateformes e-santé au sein de multiples intervenants. En outre, l'architecture de blockchain avancée suit la méthodologie des audits primaires par les parties prenantes telles que les compagnies d'assurance, les hôpitaux, et les médecins en ce qui concerne l'authenticité et la crédibilité d'un dossier qui est partagé sur la plate-forme.

Liang et al. [50], intègre un réseau Blockchain pour le Partage des données et de la collaboration avec l'architecture Blockchain. Le système décrit en général l'interopérabilité d'un groupe d'entités comme patient, médecin, fournisseurs de soins de santé et assurance les entreprises pour le partage et la collaboration des données. L'appareil portable fixé sur le patient est relié vers la base de données en nuage (Cloud) ou le réseau qui stocke l'ensemble des données du patient. Comme chaque jour une énorme quantité de données vient dans le stockage pour un traitement efficace des données et gestion de l'intégrité des données, le travail propose de former des lots de données à stocker dans une architecture arborescente Merkle et le traitement des données en douceur.

3 Cloud :

Le Cloud Computing est un modèle permettant d'établir un accès à la demande en réseau vers un bassin partagé de ressources informatiques configurables (Figure 4). 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 [51]. Toutes ces notions sont développées dans les sections suivantes.

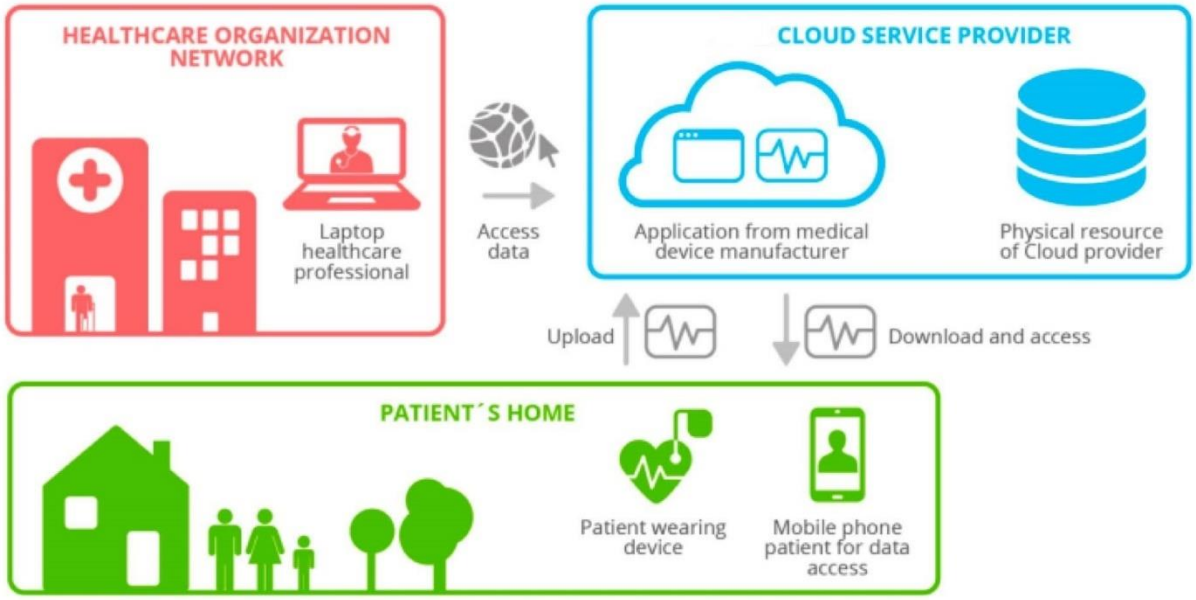

Figure 4 : Utilisation cloud dans les applications E-santé.

3.1 Caractéristiques du cloud :

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

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.

3.2 Modèles de livraisons

Il existe trois modèles de livraison du Cloud Computing :

- 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.

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.

3.3 Modèles de déploiement :

Il existe 4 modèles de déploiement de cloud qui sont :

a. Cloud privé
Un cloud privé est utilisé par une seule organisation, qui peut être gérer par l'organisation elle-même ou un partie tier, l'infrastructure de ce dernier peut être dans les locaux de l'organisation ou en dehors sous contrat avec des providers.

b. Cloud communautaire
Un cloud communautaire est partagé par plusieurs organisations pour remplir les besoins de la communauté qui veut mettre en place des moyens commun de (sécurité, stockage, etc..). Elle peut être gérée par la communauté ou par une tierce partie et peut être placée dans les locaux ou en dehors sous contrat avec des providers.

c. Cloud publique
Un Cloud publique est ouvert au public ou à des groupes industriels. Cette infrastructure est possédée par une organisation qui vend des services Cloud.

d. Cloud hybride

Ce Cloud est composé d'un ou plusieurs modèles (privée, communautaire ou publique) qui restent séparées. Ces infrastructures sont liées les unes aux autres par une technologie qui autorise la portabilité des applications et des données.

3.4 Utilisation du cloud avec blockchain :

Le Blockchain est un système de stockage en nuage décentralisé qui assure la sécurité des données. Tout nœud informatique connecté à Internet peut rejoindre et former un réseau de pairs, maximisant ainsi l'utilisation des ressources. Blockchain est un système peer to peer distribué où chaque nœud du réseau stocke une copie du blockchain ce qui le rend immutable mais crée une redondance dans les données du patient et consomme ainsi un grand espace de stockage. Pour régler ce problème de stockage, le cloud semble une bonne solution : le données médicales seront stockées dans le cloud et le blockchain garde trace des transactions (demande ajout et modification), de l'emplacement des données dans le cloud, voire le hash des données stockées et autres. En effet, le stockage cloud est l'une des principales options pour stocker des données massives (données médicales). Cependant, cette approche de stockage n'est pas sécurisée : les données sont en claires et toute personne peuvent y accéder. Pour renforcer la sécurité du cloud, nous devons utiliser d'autres mécanismes comme le contrôle d'accès et le chiffrement de données. Notre solution consistera à combiner ces technologies et techniques de sécurité : blockchain, cloud, contrôle d'accès et chiffrement dans une plateforme d'E-santé. Cette solution sera expliquée dans le chapitre 3.

4. Conclusion

Compte tenu de l'exactitude, de la sécurité et de l'authenticité des données, il est nécessaire que les données soient toujours gérées de façon appropriée. Pour cela, nous allons utiliser le concept de blockchain privé qui est un consortium de multiples intervenants tels que les Hôpitaux, Médecins, Pharmacie, Pathologie, centres d'imagerie, centres de recherche médical qui assure l'intégrité et la traçabilité. Nous allons aussi utiliser un cloud pour le stockage des dossiers médicaux. Et nous avons choisi de renforcer la sécurité du cloud par un contrôle d'accès et un chiffrement des données pour assurer la confidentialité. Ces concepts de chiffrement et de contrôle d'accès font l'objet du chapitre suivant.

CHAPITRE 2 : Chiffrement et Contrôle d'Accès

Pour assurer la sécurité dans les applications e-santé, il faut utiliser différentes techniques et technologies comme mentionné le chapitre précèdent. Le blockchain avec le cloud ne permet pas d'assurer à la fois et avec un niveau élevé la disponibilité, la confidentialité et l'intégrité des données médicales. Il faut ajouter d'autres mécanismes de sécurité comme le contrôle d'accès et le chiffrement. Ces mécanismes seront expliqués dans les trois parties de ce chapitre.

1 Cryptologie

La cryptologie est un mot d'origine Grèce, qui se compose de deux parties la « crypto » et « logo » qui signifie « cacher » et « mot ». L'art de caché ou crypté ses messages (cryptologie) existe depuis l'invention de l'écriture elle-même. L'humain par sa nature prudente et curieuse devait cacher certain message comme les communications militaires et information secret entre clan. Un exemple connu dans cette catégorie et le chiffrement de césar utilisé par l'empereur lui-même pour assurer la confidentialité de ses messages.

La cryptologie se divise en deux catégories :

1. La cryptographie : qui est l'art de changer l'information de façon à la rendre complètement illisible ou seules les personnes détenant la clé peuvent lire cette information.

2. La cryptanalyse : qui est l'art de brisé ou déchiffré l'information chiffré en utilisant différentes techniques sans connaitre la clé de chiffrement.

Dans ce qui suit, nous nous intéressons à la cryptographie.

1.1 La cryptographie :

Comme déjà cité, la cryptographie ou le chiffrement est l'art de chiffrer, coder et transformer les messages en utilisant des formules et des calculs mathématiques dans des algorithmes informatiques. Cet art a été créé par les besoins de l'humanité à garder certaine information secrète dans les temps difficile ou tout simplement pour le confort d'avoir l'information sécurisé avec une couche supplémentaire. Elle est utilisée pour dissimuler ou éviter que certain message ne soit à la portée de certains utilisateurs malveillant ou curieux. De nos jours, la cryptographie est utilisée dans toutes les actions entreprises dans notre vie quotidienne que ça soit la sauvegarde des mots de passe ou informations personnels et bancaire ou même la vérification de l'intégrité et authenticité de certaine information.

1.2 Techniques de chiffrement :

La cryptographie ou le chiffrement est un ensemble de fonctions mathématique utilisée pour le cryptage et le décryptage. Ces algorithmes sont associés à des clés[1], afin de crypter l'information. Avec des clés différentes, la sécurité de cette information dépend sur :

- L'invulnérabilité de l'algorithme de cryptographie
- La confidentialité de la clé.

Il existe trois techniques de chiffrement, à savoir le chiffrement symétrique (à clé secrète), le chiffrement asymétrique (à clé publique et privée) et les fonctions de hachage (sans clé).

1.2.1. Chiffrement Symétrique :

Les algorithmes à clef secrète ou algorithmes symétriques, ou communément appelé **Chiffrement symétrique ou Clé secrète** : utilisent une seule et même clé pour le chiffrement et déchiffrement de l'information (figure (5). Cette clé doit être stockée de façon sécurisée tel que seulement l'émetteur et le récepteur ont accès.

Figure 5 : Principe du chiffrement Symétrique [52]

Le chiffrement symétrique se divise en deux catégories :

- Chiffrement par bit, qui interagit avec l'information bit par bit. Quelques exemples sur les algorithmes de chiffrement : RC4, E0, …
- Chiffrement par block, interagit avec l'information par groupe de bits appelé block. Quelques exemples sur les algorithmes de chiffrement :
  DES, AES, …

[1] La clé est la valeur utilisée par certain algorithme de cryptographie, pour chiffrer l'information. La sécurité de la clé est primordiale ; seul le propriétaire doit en avoir accès.

1.2.2. Chiffrement asymétrique :

Ou plus communément appelée chiffrement à clé publique, ce type de chiffrement utilise 2 clés **(public, privé)** différentes pour le chiffrement et le déchiffrement de l'information qui ne peuvent être déduite en utilisant l'une des clés. C'est pour cela que ce chiffrement est appelé chiffrement à clé public où on peut partager la clé publique sans répercussion sur la confidentialité de la clé privé.

On utilise la clé publique pour chiffrer l'information qui sera déchiffré en utilisant la clé privée (figure 6). Exemples sur le chiffrement asymétrique : RSA, ElGamel, ECC

Figure 6: Principe du chiffrement Asymétrique[52]

Comparaison entre le chiffrement symétrique et asymétrique :

L'un des premiers avantages de l'utilisation du chiffrement à clé public est d'avoir un niveau de sécurité plus élevé car on a plus besoin d'utiliser différentes techniques pour envoyer la clé secrète de façon sécurisée, car on aura qu'à partager la clé publique et garder la clé privée dans endroit sure sans avoir besoin de la partager.

Aux contraires en utilisant le chiffrement à clé secrète (symétrique), il y'aura toujours le risque d'avoir la clé interceptée au cours de son transfert par un tiers qui pourra ensuite utiliser cette dernière pour modifier ou falsifier les informations à sa guise. La transmission sécurisée de la clé est le problème majeur pour tous les algorithmes de chiffrement symétrique.

Le tableau suivant résumé des avantages et inconvénients de ces derniers

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 3 : Avantages et inconvénients de la cryptographie symétrique/asymétrique

1.2.3 Fonctions de Hachage

Une fonction de hachage cryptographique est une procédure bien définie qui accepte un bloc de données et génère une chaîne de bits de longueur fixe appelée valeur de hachage ou digest.

Une bonne fonction de hachage cryptographique est définie par ses propriétés, y compris [53]:

• Une fonction de hachage cryptographique est déterministe, de sorte qu'un bloc donné de données d'entrée générera la même valeur de hachage, peu importe combien de fois il est exécuté à travers la même fonction de hachage.

• Une fonction de hachage est unidirectionnelle, ce qui signifie que la procédure de la fonction de hachage est irréversible. Il n'y a pas de fonction pour dériver les données de la source simple à partir de la valeur de hachage.

• Un petit changement dans les données sources devrait générer un changement substantiel dans la valeur de hachage, Ceci est connu comme un effet en cascade.

•Une fonction de hachage doit être sans collision, de sorte que les chances de deux blocs de données sources différents générant la même valeur de hachage devraient être extrêmement peu probable. Une collision de hachage général peut être générée par données aléatoires. C'est souvent le type de hachage que les pirates informatiques cherchent quand ils essaient d'accéder aux systèmes sécurisés où les hachures des mots de passe de connexion sont stockées, au lieu des mots de passe réels eux-mêmes.

Actuellement, il existe plusieurs des fonctions de hachage : SHA-256, MD5,MD4, SHA1, NTLM, Keccak-256. La fonction de hachage la plus utilisée et connue est SHA-256 que nous utiliserons aussi dans notre travail.

2 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. Dans cette partie nous parlerons des différents modèles de contrôle d'accès.

2.1 Modèles de contrôle d'accès discrétionnaires (DAC)

Le contrôle d'accès discrétionnaire (DAC) a été défini à l'origine par Trusted Computer System Evaluation Criteria (TCSEC) comme « un moyen de restreindre l'accès aux objets en fonction de l'identité des sujets et/ou des groupes auxquels ils appartiennent. Les contrôles sont discrétionnaires en ce sens qu'un sujet ayant une certaine permission d'accès peut transmettre cette permission (peut-être indirectement) à tout autre sujet (à moins qu'elle ne soit restreinte par un contrôle d'accès obligatoire). »

En pratique, l'utilisation de cette terminologie n'est pas aussi claire. Dans l'interprétation la plus stricte, chaque objet contrôlé par un DAC doit avoir un propriétaire qui contrôle les permissions permettant l'accès à l'objet (figure 7). Bien que de nombreux systèmes d'exploitation modernes prennent en charge le concept de propriétaire, cela n'est pas toujours mis en œuvre. En particulier, la norme ne couvre pas les « propriétaires », ce qui laisse une définition problématique lorsque la propriété de groupe se produit.[54]

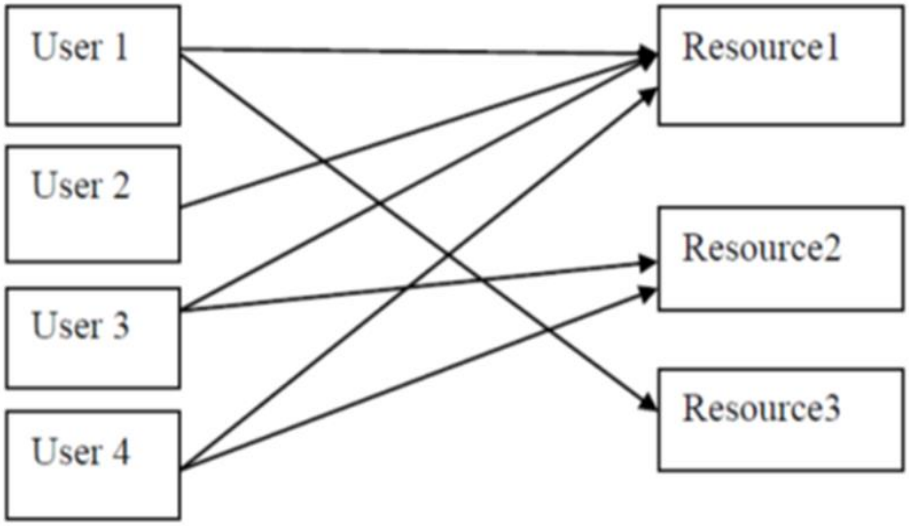

Figure 7: modele de controle d'acces DAC[55]

2.2 Modèles de contrôle d'accès obligatoires (MAC)

Le Mandatory Access Control imposent le contrôle d'accès sur la base de la réglementation mandatée par une autorité centrale. La forme la plus courante de cette politique est la politique de sécurité à plusieurs niveaux, basée sur la classification des sujets et des objets dans le système(figure 8). Les objets sont des entités passives stockant des informations. Les sujets sont des entités actives qui demandent l'accès aux objets. Notez qu'il y a une distinction entre les sujets de la politique obligatoires(mandatory) et les sujets d'autorisation politiques discrétionnaires. Bien que l'autorisation correspondent aux utilisateurs (ou groupes de ceux-ci), les politiques obligatoires(mandatory) font une distinction entre les utilisateurs et les sujets. Les utilisateurs sont des êtres humains qui peuvent accéder au système, tandis que les sujets sont des processus (c.-à-d. des programmes en cours d'exécution) fonctionnant pour le compte de utilisateurs. Cette distinction permet à la politique de contrôler les accès indirects (fuites ou modifications) causées par l'exécution de processus.[56]

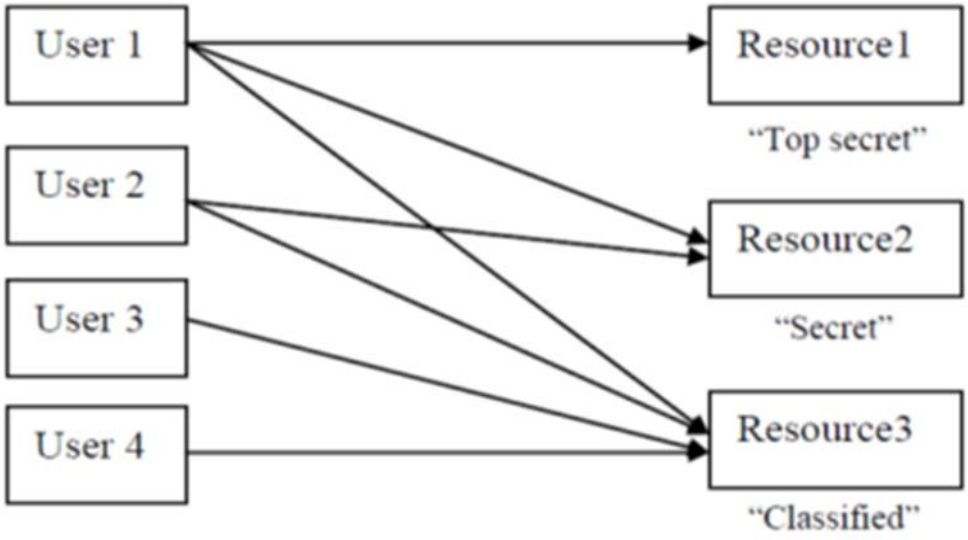

Figure 8 : modele de controle d'acces MAC [55]

2.3 Modèles de contrôle d'accès à base de rôles (RBAC)

Les modèles DAC et MAC ne sont pas bien adaptés aux besoins des organisations commerciales. Dans ce type d'organisation les privilèges conférés aux utilisateurs dépendent du rôle des utilisateurs au sein de l'organisation. De ce fait les modèles RBAC sont apparus et se sont imposés comme une alternative aux modèles DAC et MAC traditionnels. En 2004, l'International Committee for Information Technology Standards de l'American National Standards Institute (ANSI/INCITS) a officiellement élevé au statut de standard la proposition de Sandhu, Ferraiolo and Khun [57]. Les principes de base du modèle RBAC sont les suivants:

- Alors que dans les modèles DAC, les permissions ont trait à des opérations de bas niveau telles que les opérations de lecture/écriture, dans les modèles RBAC elles concernent des tâches de nature organisationnelle telles que « transférer de l'argent », « acheter un billet d'avion » etc voir (figure 9).

- Dans les modèles RBAC, le concept de rôle correspond à une fonction professionnelle. Les permissions sont accordées à des rôles et non pas à des utilisateurs. Les rôles sont ensuite distribués aux utilisateurs en fonction de leurs responsabilités au sein de l'organisation. Une même permission peut être affectée à différents rôles et différents rôles peuvent être attribués à un même utilisateur.

- Les modèles RBAC offrent une solution pour implanter des mesures de type séparation des tâches. Le principe de la séparation des tâches prévoit qu'un même utilisateur ne peut effectuer des tâches qui pourraient être orchestrées pour mettre oeuvre des opérations frauduleuses, comme par exemple « autoriser un paiement » et « effectuer un paiement ». Ce principe peut aisément être garanti avec les modèles RBAC dans la mesure où deux rôles peuvent être déclarés comme étant mutuellement exclusifs. Deux rôles mutuellement exclusifs ne peuvent alors être affectés à un même utilisateur.

Le standard RBAC ne dit rien au sujet de l'administration du règlement de sécurité. Il suppose de manière implicite que la définition des rôles, l'affectation des permissions aux rôles et la distribution des rôles aux utilisateurs sont effectuées par une autorité centrale. Le modèle ARBAC (Administrative Role-Based Access Control est un modèle à base de rôles prévoyant des rôles correspondant aux fonctions d'administration du règlement de sécurité. Les modèles à base de rôles ont été implantés dans de nombreux systèmes et applications tels que Microsoft Active Directory, la plupart des SGBD commerciaux, FreeBSD et Wikipedia[58].

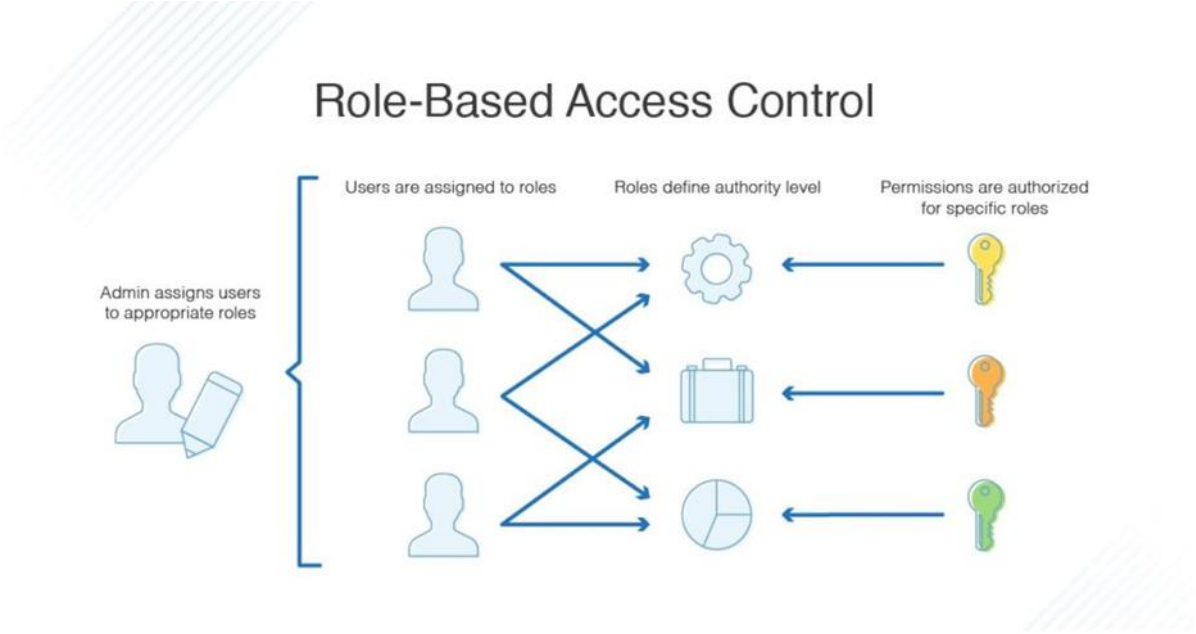

Figure 9 : modele de controle d'acces RBAC[59]

2.4 Modèles de contrôle d'accès à base d'attribut (ABAC)

Le modèle ABAC a été développé par Eric Yuan et Jin Tong dans le but de pallier aux difficultés que rencontrent les architectures web services en termes de sécurité. En effet, les accès à l'information au niveau de ces architectures web services se font non seulement sur les systèmes distribués mais très dynamiquement. Les modèles classiques sont généralement destinés à un fonctionnement statique, ils ne permettent guère une évolution dynamique. De part sa définition, le modèle ABAC peut être le plus adapté pour les architectures fonctionnant dans un environnement ouvert « in the cloud » où différentes organisations peuvent assurer à la fois les accès aux informations et la protection de leurs ressources.

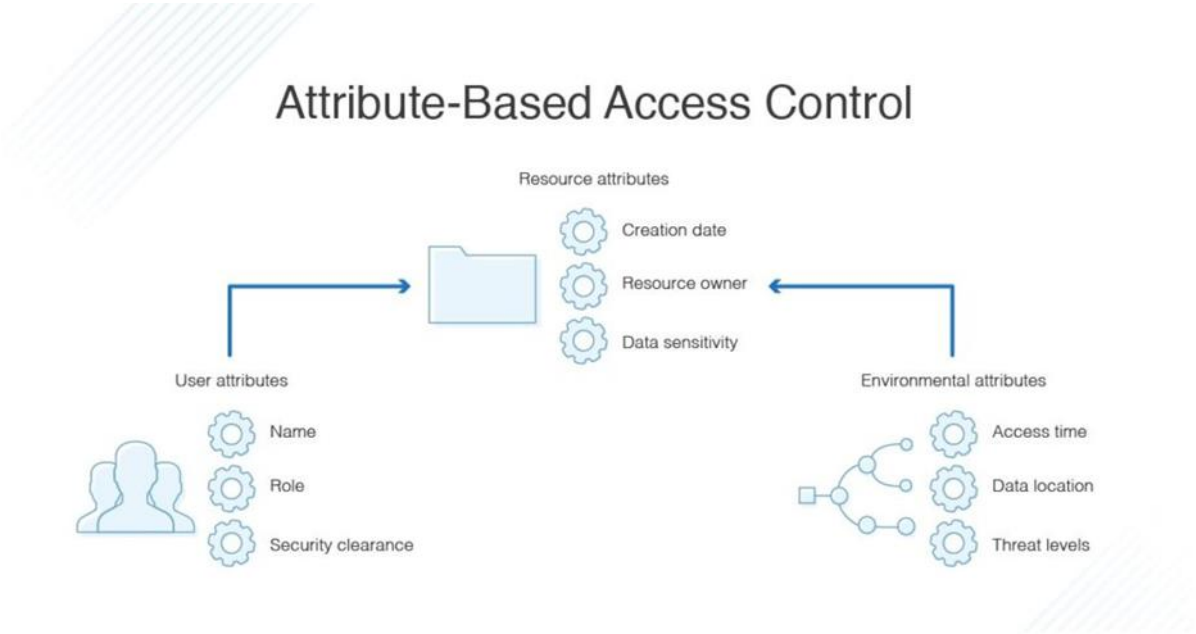

Figure 10 : modele de controle d'acces ABAC[59]

Comme son nom l'indique, le modèle ABAC définit les autorisations d'accès en se basant sur des caractéristiques de chaque entité, appelés attributs (figure 10). Trois groupes d'attributs se distinguent selon le type de l'entité à laquelle ils s'appliquent :

1.Les attributs des sujets : un sujet est une entité qui peut agir sur une ressource. A chaque sujet on associe des attributs qui définissent son identité et ses caractéristiques. Par exemple le rôle du sujet peut aussi être considéré comme un attribut, tout comme le nom, le prénom, ou le titre, etc.

2.Les attributs des ressources : C'est un objet du système sur lequel un sujet peut agir. Autrement dit, c'est une entité qui peut être accéder accessible à un sujet. Une ressource peut être un fichier, un service, etc. A chaque ressource est associée des attributs, variables selon sa nature, mais qui peuvent être : son type, le nom de son auteur, son propriétaire, la date de modification, etc.

3.Les attributs d'environnement : l'environnement peut être décrit par des informations opérationnelles, techniques, liées à la situation ou encore au contexte dans lequel l'accès à l'information se produit. La particularité du modèle, ABAC est la prise en compte du contexte d'exécution du système, en définissant des attributs d'environnement, comme par exemple : la date, le niveau de sécurité du réseau, le débit de la connexion, etc.

Cette particularité de prise en compte du contexte est très importante dans notre problématique où nous sommes appelés à gérer des tiers sujets n'appartenant à aucune organisation et qui souhaitent obtenir un service auprès de tierces organisations.

Un des avantages de ce modèle est d'exploiter les opportunités offertes par les autres modèles, par exemple avec l'attribut du sujet « rôle » on peut appliquer le modèle RBAC.

2.5 Comparaison entre les différent modèles de contrôle d'accès

Le tableau 6 présente les avantages et les inconvénients des modèles de contrôle d'accès décrits précédemment. Notons ici que

- Le contrôle d'accès discrétionnaire et obligatoire offre deux extrémités opposées du spectre. Bien qu'un modèle de DAC puisse convenir aux petites entreprises, il ne favorise certainement pas l'expansion ni la sécurité de l'environnement à mesure que l'entreprise se développe.
- Un modèle MAC pourrait être assez sûr, mais il vient avec un coût opérationnel élevé, et en tant que tel est la méthode la moins efficace. Il ne faut pas oublier que la sécurité est une question d'équilibre. D'un coté, on veut donner aux gens le moyen de faire leur travail, sans sacrifier la sécurité. D'un autre côté, on ne veut pas verrouiller les choses jusqu'à ce que personne ne puisse faire son travail.
- Une distinction clé entre RBAC et ABAC est leur nature statique par rapport à dynamique, comme implicite dans leurs modèles respectifs. RBAC permet l'accès basé sur des rôles, qui sont généralement assez statiques au sein d'une organisation. Par ailleurs, ABAC s'appuie sur des attributs, qui peuvent être dynamiques - changeants, par exemple, lorsqu'un utilisateur tente d'accéder à une ressource à partir d'un périphérique ou d'une adresse IP différent.
- ABAC peut être automatisé pour mettre à jour les autorisations et, une fois que tout est configuré, nécessite moins d'administration globale. Il est également sécurisé lorsqu'il est correctement configuré. Toutefois, ABAC peut être assez complexe et spécifique à l'environnement, et les ensembles d'attributs compliqués peuvent être difficiles à mettre à l'échelle. Il est également difficile d'effectuer un audit à des fins de conformité - on doit vérifier chaque objet individuel par rapport à notre politique d'accès, au lieu de simplement vérifier l'accès d'un utilisateur particulier.
- RBAC, d'autre part, est très efficace et peut rationaliser le processus de conformité. Bien que toute forme de contrôle d'accès s'accompagne d'un certain degré de complexité, le RBAC est suffisamment transparent pour qu'on puisse voir comment les individus interagissent avec les ressources en fonction de leurs rôles. Et, conformément à l'adage : «La complexité est l'ennemi de la sécurité», parce que sa configuration est relativement simple, on peut contrôler plus facilement l'accès aux données sensibles, ce qui peut entraîner moins de violations. Cependant, la gestion des rôles de RBAC peut devenir difficile et complexe dans un environnement qui a une multitude de rôles différents, chacun avec son propre ensemble complexe d'autorisations.

Dans notre projet, MAC et DAC ne sont pas des choix judicieux que l'on pourrait utiliser dans une application e-santé basé sur le blockchain. Par ailleurs, les modèles ABAC et RBAC semblent plus réalisables et sont les plus utilisés dans ce domaine. De plus, ABAC avec son utilisation de multiple et différent type d'attributs (environnement, ressources, sujets) qui définisse l'utilisateur et avec la possibilité d'utilisation du chiffrement par attribut (ABE), semble beaucoup plus adapté pour les applications e-santé.

contrôle d'acces	Les avantages	Les inconvénients
DAC	**user-friendly :** Les utilisateurs peuvent gérer leurs données et accéder rapidement aux données d'autres utilisateurs. **Flexible** : Les utilisateurs peuvent configurer les paramètres d'accès aux données sans administrateur. **Facile à entretenir :** L'ajout de nouveaux objets et d'utilisateurs ne prend pas beaucoup de temps pour l'administrateur. **Granulaire :** Les utilisateurs peuvent configurer les paramètres d'accès pour chaque élément de données.	**Faible niveau de protection des données** : Le DAC ne peut garantir une sécurité fiable car les utilisateurs peuvent partager leurs données comme ils le souhaitent. **Obscur :** Il n'y a pas de gestion centralisée des accès, donc pour connaître les paramètres d'accès, on doit vérifier chaque ACL.
MAC	**Niveau élevé de protection des données :** Un administrateur définit l'accès aux objets, et les utilisateurs ne peuvent pas modifier cet accès. **Granulaire** : Un administrateur définit manuellement les droits d'accès des utilisateurs et les paramètres d'accès aux objets. **Immunisé contre les attaques de chevaux de Troie :** Les utilisateurs ne peuvent déclassifier les données ou partager l'accès aux données classifiées.	**Maintenabilité:** La configuration manuelle des niveaux de sécurité et des habilitations nécessite une attention constante de la part des administrateurs. **Évolutivité :** MAC ne s'adapte pas automatiquement. **Non convivial :**

		Les utilisateurs doivent demander l'accès à chaque nouvelle donnée; ils ne peuvent pas configurer les paramètres d'accès pour leurs propres données.
RBAC	**contrôle précis :**plus besoin d'autoriser ou de révoquer l'accès sur une base individuelle, rassemblant les utilisateurs en fonction de leurs rôles à la place **facile à implémenter :** L'établissement d'un ensemble de rôles dans une petite ou moyenne entreprise n'est pas difficile	**Statique :**Les permissions ne peuvent être attribuées qu'aux rôles des utilisateurs et non aux objets et aux opération
ABAC	**Dynamique :** accorde l'accès en fonction non pas du rôle de l'utilisateur, mais des attributs de chaque composante du système. De cette façon, on peut décrire une règle d'affaires de toute complexité. Même si on doit rendre certaines données accessibles uniquement pendant les heures de travail	ce type de système est difficile à configurer en raison de la façon dont les politiques doivent être spécifiées et maintenues

Tableau 4 : Comparaison entre les différents modèles de contrôle d'accès

2.6 Le contrôle d'accès par la cryptographie (ABE) :

Le modèle de contrôle d'accès par la cryptographie se repose sur l'utilisation des techniques de chiffrement pour renforcer le contrôle d'accès.

Le modèle ABE a été proposé par Sahai et Waters en 2005. ABE permet aux utilisateurs de chiffrer et de déchiffrer des données en fonction d'attributs d'utilisateur. La clé secrète d'un utilisateur et le texte chiffré dépendent d'attributs voir (figure 11). Le déchiffrement d'un texte crypté n'est possible que si l'ensemble des attributs de la clé d'utilisateur correspond aux attributs du texte crypté. ABE applique le contrôle d'accès via la cryptographie à clé publique. L'objectif principal de ces modèles est d'assurer la confidentialité et le contrôle d'accès. Les principaux aspects sont la flexibilité, l'évolutivité et un contrôle d'accès à granularité fine.[60]

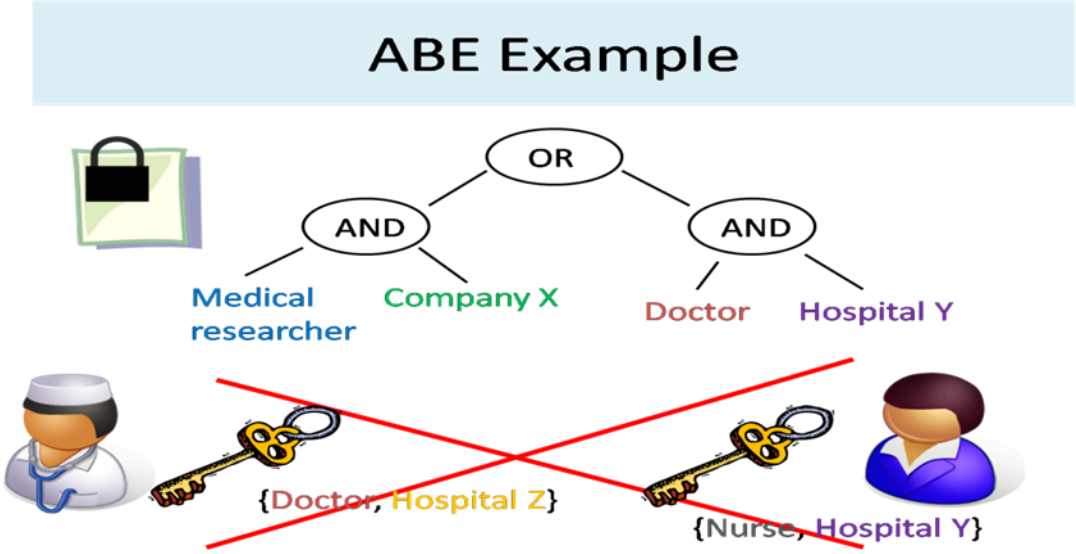

Figure 11 : contrôle d'accès par la cryptographie

Nous remarquons que le problème du contrôle d'accès se pose au niveau du Cloud qui doit être de confiance, d'autres solutions se proposent de résoudre ce problème. L'approche proposée par M. Li et al [44] est basée sur Attribute Based Encryption (ABE) pour contrôler l'accès aux données médicales hébergées dans le Cloud. Cette solution a la particularité de renforcer la protection de la vie privée en réduisant les privilèges de l'hébergeur des données.

3 Le Chiffrement par attributs

Traditionnellement, un schéma de chiffrement comme RSA et AES fournit une transmission et un stockage de données sécurisées dans un environnement Cloud mais le problème 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. Une solution à ce problème est donnée par le chiffrement par attributs ou « Attribute Based Encryption» (ABE) [61], 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 des fonctionnalités de chiffrement et de contrôle d'accès. Le chiffrement par attributs (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 de plusieurs clés pour déchiffrer. Un avantage évident de cette technique est que chaque utilisateur à une clé dédiée. Aussi bien, ABE fournit un contrôle d'accès à forte granularité [61]. 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.

Les critères des schémas de chiffrement idéaux basés sur les attributs, sont listés comme suit [62]:

- *C1. Confidentialité des données:* Avant de stocker des données dans le nuage, celles-ci doivent être cryptées par le propriétaire des données. Par conséquent, les parties non autorisé, y compris le nuage, ne peuvent pas connaître des informations sur les données cryptées.
- *C2. Contrôle d'accès avec une fine granularité:* Dans le même groupe, le système a accordé de différent droit d'accès à l'utilisateur individuel. Les utilisateurs sont sur le même groupe, mais chaque utilisateur peut se voir attribuer le droit d'accès aux données. Même pour les utilisateurs dans le même groupe, leurs droits d'accès ne sont pas les mêmes.
- *C3. L'évolutivité :* Lorsque le nombre d'utilisateurs autorisés augmente, le système peut travailler efficacement. Donc, le nombre d'utilisateurs autorisés ne peut pas affecter les performances du système.
- *C4. Responsabilité de l'utilisateur :* Si l'utilisateur autorisé est malhonnête, il partagerait son attribut clé privée avec l'autre utilisateur non autorisé. Cela cause le problème que la clé illégale serait partager entre utilisateurs non autorisés.
- *C5. Révocation d'utilisateur :* Si l'utilisateur quitte le système, le système peut révoquer son accès depuis le système directement. L'utilisateur révocable ne peut accéder aux données stockées, car son droit d'accès a été révoqué.
- *C6. Résistant à la collusion :* Les utilisateurs ne peuvent pas combiner leurs attributs pour déchiffrer les données cryptées. En effet, une bonne construction d'ABE ne doit pas permettre à deux utilisateurs de combiner leurs clés privées pour déchiffrer des données pour laquelle ils n'ont pas d'accès individuel, en d'autres termes, deux utilisateurs ne doivent pas pouvoir combiner leurs attributs pour pouvoir avoir des droits d'accès supérieurs à ceux qu'ils ont individuellement.

3.1 Les variantes principales du chiffrement basé sur les attributs (ABE) :

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 (figure 12). Les deux principales variantes sont « Ciphertext-Policy Attribute Based Encryption » (CP-ABE) proposée pour la première fois par J. Bethencourt et al [63], dans CP-ABE, la politique d'accès est intégrée dans le texte chiffré et les clés secrètes sont générées avec un ensemble d'attributs décrivant l'utilisateur légitime qui pourra déchiffrer ce texte. Seule la clé secrète avec un ensemble d'attributs qui satisfait la politique d'accès précédente peut récupérer le texte clair. La deuxième variante est« Key-Policy Attribute Based

Encryption » (KP-ABE) [32]. Pour KP-ABE, la politique d'accès est intégrée dans la clé secrète, en d'autres termes, on décide pour chaque utilisateur quels sont les objets auxquels il aura accès. On attache à chaque texte chiffré un ensemble d'attributs. Une clé secrète donnée, avec une politique d'accès donnée, ne peut déchiffrer que le texte chiffré ayant les attributs qui satisfont sa politique d'accès.

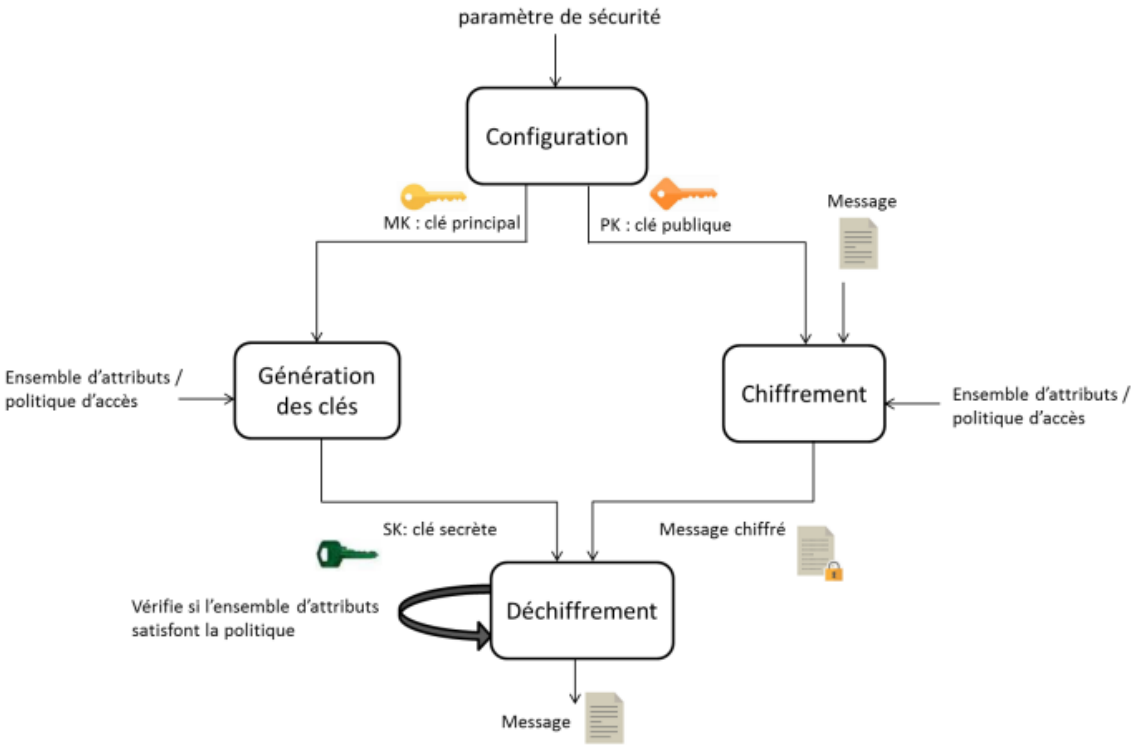

Figure 12: algorithme ABE

3.1.1. Algorithme KP-ABE (Key-Policy Attribute-Based Encryption):

Goyal et al [33] ont proposé un schéma de cryptage basé sur les attributs de stratégie de clé (KP-ABE) qui intégrait la stratégie d'accès à la clé privée de l'utilisateur et définissait les données cryptées avec les attributs d'utilisateur. Le système KP-ABE a le potentiel d'obtenir un contrôle d'accès fin. Dans KP-ABE, le contrôle des utilisateurs est plus flexible que le système ABE traditionnel. Ainsi, KP-ABE est une forme améliorée de Sahai et Waters ABE [64].

Néanmoins, le problème avec le schéma KP-ABE est que la stratégie d'accès est ajoutée à la clé privée de l'utilisateur, en tant que tels, les propriétaires de données ne peuvent pas décider qui peut déchiffrer les données cryptées, à l'exception de la sélection d'un ensemble d'attributs utilisés pour décrire les données. La figure 13 ci-dessous montre comment Les systèmes KP-ABE fonctionnent. [65]

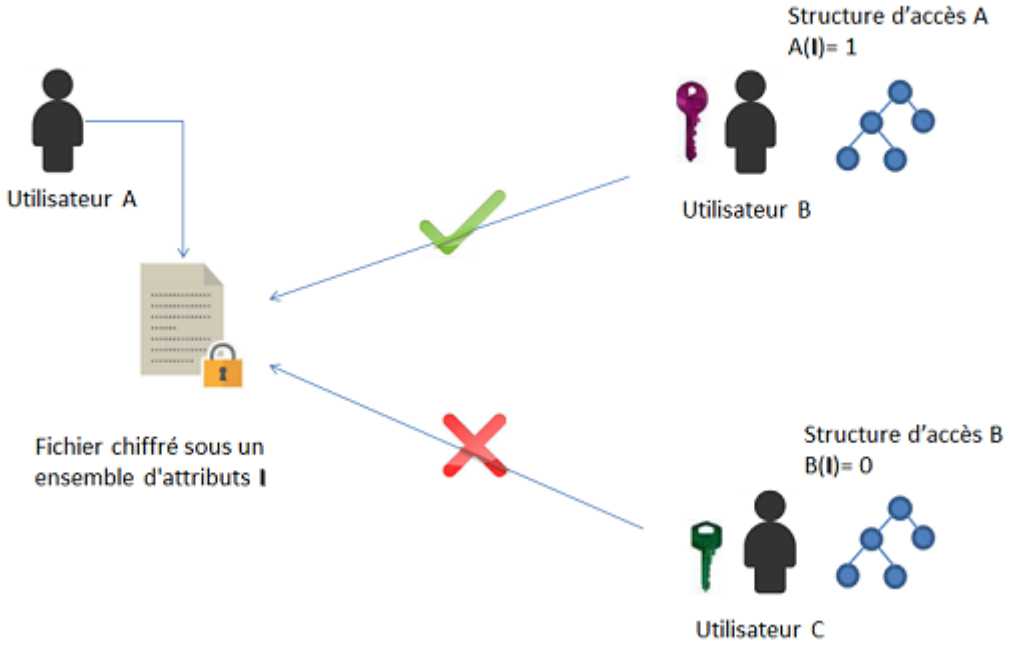

Figure 13 : Chiffrement KPABE

Dans cette figure, 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.

3.1.2. Algorithme CP-ABE (Ciphertext-Policy Attribute-Based Encryption)

Bethencourt et al [66] ont proposé un schéma de chiffrement basé sur l'attribut de politique de chiffrement (CP-ABE). Le schéma CP-ABE inclut la politique d'accès dans les données, laquelle est chiffrée avec l'ensemble des attributs de la clé de l'utilisateur. Plusieurs variantes de systèmes ABE ont été suggérées sur la base du modèle CP-ABE. [65]

Le cryptage à base d'attributs de stratégie de texte chiffré (CP-ABE) peut être généralement appliqué comme méthode de contrôle d'accès [66]. Les dossiers médicaux sensibles, étroitement liés à la vie privée des patients, ne doivent être accessibles que si les patients y consentent. Récemment, CP-ABE a démontré sa capacité à gérer efficacement le DSE en cryptant les dossiers médicaux avec des structures d'accès expressif incluent « Spécialité: Médecine» et/ou «fonction: Médecin» [65]. Les caractéristiques de CP-ABE en font un bon candidat pour sécuriser l'accès au DSE dans un environnement d'informatique en nuage et par rapport à KP-ABE, un meilleur candidat. La figure 14 ci-dessous montre comment les systèmes CP-ABE fonctionnent. [65]

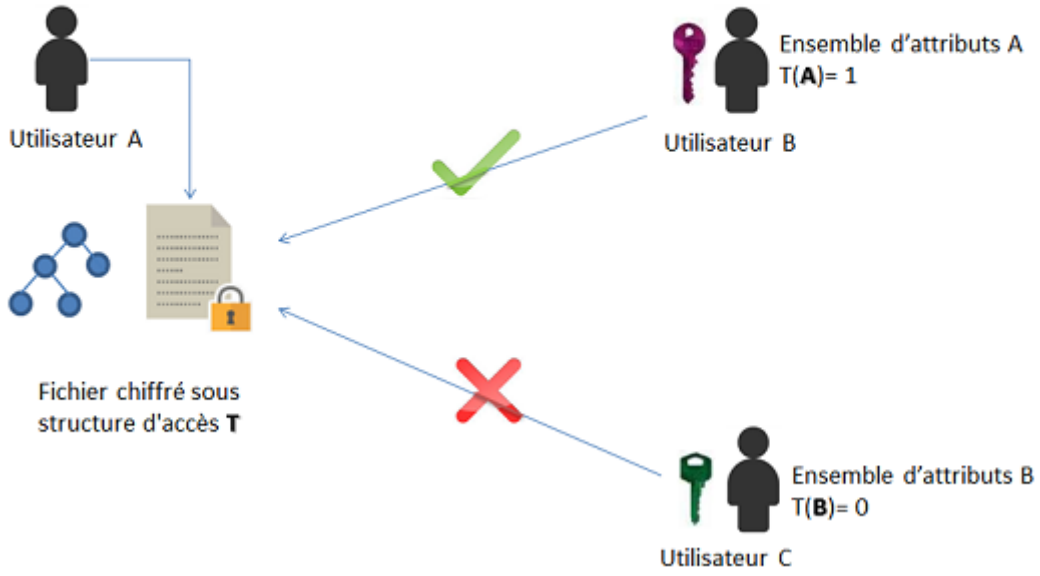

Figure 14 : Chiffrement CPABE

3.1.3. Comparaison entre CP-ABE et KP-ABE

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. [68]
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. Ainsi, le CP-ABE possède les avantages suivants : [58]:

- assure la confidentialité des données;
- Permet le Contrôle d'accès avec une fine granularité [57];
- Permet L'évolutivité;
- Résistant à la collusion;
- frais généraux de calcul raisonnables.

C'est pourquoi, nous optons pour l'utilisation de l'algorithme CP-ABE dans notre plateforme d'e-santé. Au fait, cet algorithme est déjà utilisé et appliqué dans le domaine de e-santé et le cloud comme nous le verrons dans la section suivante ;

3.2 Domaines d'application du CP-ABE :

Récemment, plusieurs protocoles de sécurité adoptent le chiffrement à base d'attributs de s (CP-ABE) en tant que bloc de construction dans différents environnements distribués, tels que systèmes médicaux et services Cloud. CP-ABE est un schéma de clé publique dans lequel le cryptage et le décryptage sont basés sur des stratégies d'accès aux données de haut niveau compte tenu des exigences susmentionnées.

3.2.1 Domaines d'e-santé (systèmes DSE):

CP-ABE a démontré sa capacité à gérer efficacement les DSE en cryptant les dossiers médicaux avec des structures d'accès expressifs. La stratégie d'accès est associée au message chiffré et la clé privée de l'utilisateur est spécifiée par un ensemble d'attributs. En tant que tels, les propriétaires de données peuvent décider qui peut déchiffrer les données cryptées. En ce sens, on pourrait dire que CP-ABE est le mieux adapté pour les systèmes DSE. Dans ce qui suit, nous citons deux travaux dans ce domaine :

- -Changji Wang et al [69] décrivent leur travail « Implementing a PersonalHealth Record Cloud Platform Using Ciphertext-Policy Attribute-Based Encryption » sur la conception et la mise en place d'un nuage de dossiers de santé personnels centrés sur le patient, plate-forme basée sur le système open-source Indivo X. Ils adoptent le chiffrement basé sur attributs de politique de cryptogramme (CP-ABE) pour assurer la confidentialité des données et le contrôle d'accès à granularité fine.C'est en ce sens, les mécanismes de contrôle d'accès traditionnels ainsi que les techniques traditionnels de cryptage ne sont pas adaptées à une utilisation publique des scénarios de Cloud computing DMP, qui doivent garantir la confidentialité, protection et contrôle d'accès à granularité fine. Le schéma CP-ABE a montré qu'il est plus utile dans un environnement e-santé puisque la politique d'accès est appliquée en associant la politique de contrôle d'accès aux données protégées. Cela supprime la nécessité d'impliquer une entité de confiance qui doit appliquer des politiques d'accès. [69]
- -Suhair Alshehri et al [70] ont présenté une conception « Designing a Secure Cloud-Based EHR System using Ciphertext-Policy Attribute-Based Encryption » pour un système de DSE basé cloud sécurisé en utilisant CP-ABE qui fournit des solutions efficaces à certains problèmes liés aux mécanismes de cryptage standard. Ils sont également étudié la faisabilité de l'adoption CP-ABE en termes de performances et de surcharge de stockage. Pour évaluer la faisabilité de leur proposition, ils ont effectué plusieurs expériences préliminaires pour mesurer les frais généraux du temps et du stockage.Les résultats suggèrent que la conception proposée a fourni des performances raisonnables et consomme un stockage négligeable, et ainsi, il peut être utilisé en remplaçant des mécanismes de cryptage standards de gestion des systèmes de DSE basés sur le Cloud. [70]
-

3.2.2. Services Cloud :

Le chiffrement à base d'attributs de stratégie de texte chiffré (CP-ABE) pourrait être un outil cryptographique efficace pour la gestion sécurisée des données stockées dans les serveurs de stockage de données. Pour donner une idée sur cette efficacité de CP-ABE, nous citons quelques exemples :

- B. Raja et al ont présenté dans [71] une solution cryptographique basée CP-ABE prometteuse. Il permet aux propriétaires de données de définir leurs propres politiques d'accès sur les attributs de l'utilisateur et d'appliquer les politiques sur les données avant d'être distribuées. Ceci élimine la nécessité de s'appuyer sur le serveur de stockage de données pour prévenir l'intégrité et l'accès non autorisé aux données. Les analyses de performances et de la sécurité indiquent que le schéma proposé est efficace pour gérer en toute sécurité les données stockées dans les serveurs de stockage de données.
- Q Yuan et al étudient dans [72] le problème de contrôle d'accès sur des données affinées dans le Cloud computing. Fondé sur un schéma CP-ABE, ils proposent une stratégie de contrôle d'accès pour atteindre une granularité fine et mettre en œuvre l'opération de révocation d'utilisateur efficacement. Les résultats de l'analyse indiquent que ce régime assure la sécurité des données dans le Cloud computing et réduit de manière significative le coût de leur propriétaire.
- Kan Y. et al [73] ont proposé une architecture basée CP-ABE. Il sont utilisé un schéma de contrôle d'accès de données affinées où le propriétaire a été chargé de définir et d'appliquer la stratégie d'accès. Ils ont également proposé une méthode de révocation attribut efficace pour CP-ABE, qui peut réduire considérablement le coût de renonciation de l'attribut. L'analyse montre que leur système de contrôle d'accès proposée est efficace et sûre prouvée dans le modèle aléatoire d'oracle. Bien que ce travail est spécifique aux systèmes de stockage de nuage, mais il est vrai qu'on peut l'appliquer dans un cloud privé.

4. Conclusion

Au cours de ce chapitre, nous avons abordé la cryptographie le contrôle d'accès et le Chiffrement par attributs, où l'accent a été mis sur le cryptage à base d'attributs CP-ABE, que nous allons utiliser pour assurer le contrôle d'accès et le chiffrement de nos données sur le cloud. Notre solution sera expliquée dans le chapitre suivant.

CHAPITRE 3 : Conception de la solution

Dans les applications e-santé, la sécurité des données médicales des patients est considérée comme une priorité lors de la conception de tel système. Pour arriver à achever un niveau de sécurité élevé, nous utilisons le blockchain une des nouvelles technologies P2P qui a fait beaucoup de bruit ses dernière années, ainsi que le cloud pour le stockage des dossiers médicales et le CP-ABE pour le contrôle d'accès et le cryptage de ces derniers. Dans ce chapitre, nous allons expliquer notre proposition qui combine ces techniques de sécurité en commençant par une description générale de notre solution. Ensuite, nous détaillons l'architecture de notre blockchain, l'interaction avec le cloud et l'utilisation du chiffrement CP-ABE. Puis, nous présentons l'architecture de notre plateforme. Enfin, nous terminons avec une étude conceptuelle de notre application.

1 Description de la solution :

Pour la conception de notre application, nous sommes passés par plusieurs étapes de test et d'échec afin de trouver la solution la plus optimale et la plus logique qui assurer le plus haut niveau de sécurité. Ces étapes peuvent se résumer comme suit (figure 15) :

1. En premier temps, nous avons essayé de trouver la meilleure utilisation de la technologie blockchain dans le domaine des applications e-santé. Pour cela, nous avons exploré les blockchains publiques et s'il serait possible de les utiliser toute en assurant la sécurité des informations. Après plusieurs recherches, nous sommes arrivés à la conclusion qu'utiliser un blokchain privé serait la solution la plus optimale qui assurera le plus haut degré de sécurité et de contrôle sur l'information comme déjà expliqué dans la section 2 du chapitre 1
2. Ensuite, nous avons passé au problème du stockage des données médicales. Comme connu, ces données sont volumineuses et consomment de plus en plus d'espace mémoire. En utilisant le blockchain uniquement, nous créons une redondance des données car il existera une copie de toutes les informations (données) sur chaque nœud du réseau. Pour remédier à ce problème, nous allons utilisons un cloud qui se marie bien avec la technologie blockchain ; Ce cloud doit permettre de créer un hash des données stockées qui va être enregistré dans les blockchains. L'utilisation du blockchain avec ce cloud sera expliquée en détail par la suite.
3. Enfin, il fallait aussi contrôler l'accès aux données médicales des patients afin de garantir la confidentialité et l'intégrité de ses données dans le cloud. Après avoir fait des recherches sur les méthodes de contrôle d'accès utilisées dans les applications e-santé, nous sommes arrivés à la conclusion que le contrôle d'accès par attribut ABAC serait le plus adéquat dans notre plateforme. Ce modèle est réalisé à l'aide du chiffrement CP-ABE.

La description de comment nous allons combiner et faire fonctionner ces technologies ensemble sera expliquée dans les sections suivantes.

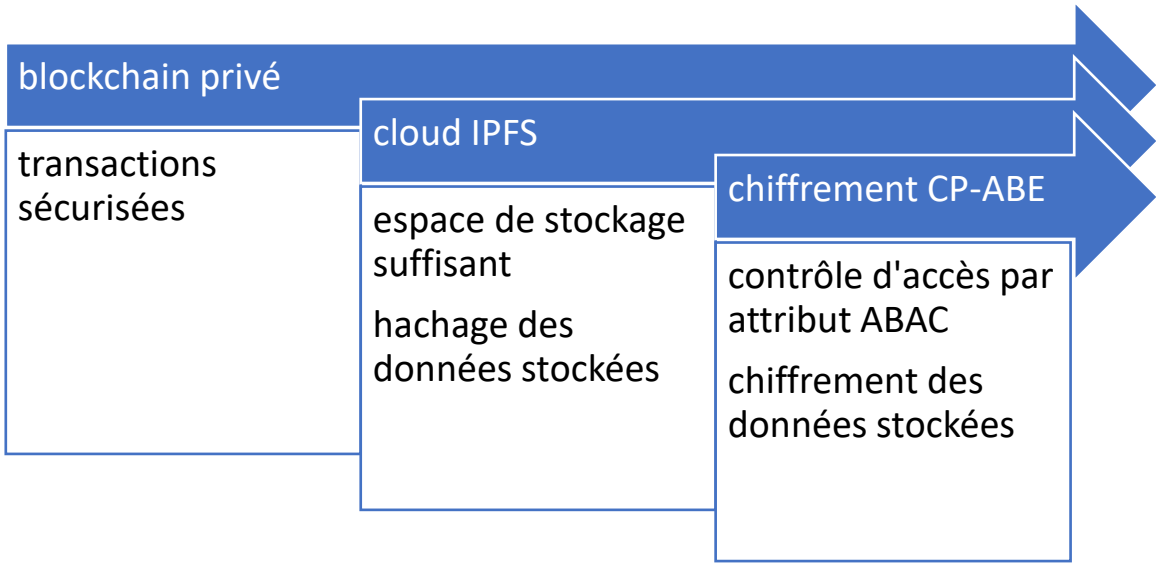

Figure 15:vue simplifiée sur notre démarche

1.1. Caractéristiques de notre Blockchain :

Dans notre application, nous avons opté pour un blockchain privé qui permet un accès limité aux données médicales où tous les participants n'ont pas accès aux données. En effet, seules les personnes (médecins ou patients) autorisés peuvent accéder et gérer leurs propres données et dossiers médicaux. Dans ce blockchain :

- L'algorithme de consensus opté est Proof Of Work « POW » vu qu'il est le plus utilisé et le plus stable (l'autre algorithme proof of stake est encore dans l'étape expérimentale donc il n'y a pas encore une version stable utilisable).
- Les nœuds sont des ordinateurs appartenant aux ministères de la santé qui peuvent se trouver dans les hôpitaux ou les cliniques médicales.
- Les transactions peuvent être :

1. Transaction de lecture : qui ne requière pas de consensus car il n'y aura pas de modification sur notre blockchain. Exemples de ces transactions :
    a. Authentification de l'utilisateur
    b. Lecture des dossiers médicaux
    c. Lecture des attributs du médecin

2. Transaction d'écriture : qui requière un consensus, car notre blockchain sera modifiée en ajoutant un block qui devra être ajouté et validé sur tous les nœuds de notre réseau blockchain. Exemple de ces transactions :
    a. Création d'un nouvel utilisateur.
    b. Modification des attributs d'un médecin.
    c. Création/modification d'un dossier.

- Les informations stockées sont dans les transactions sont : les informations d'authentification, les attributs et les droits d'accès nécessaires au chiffrement CP-ABE, l'emplacement des données dans le cloud ainsi que le hachage des données cryptées..

1.2. Stockage des données dans le Cloud

Dans notre cloud, nous utilisons le protocole IPFS(InterPlanetary File System) pour le stockage des données. Il fournit un stockage distribué peer-to-peer (P2P) structuré dans lequel de grands volumes de dossiers médicaux peuvent être facilement stockés. Le contenu est accessible par des pairs situés n'importe où dans le monde, qui peuvent relayer l'information, la stocker ou faire les deux. Au fait, chaque pair (nœud réseau) ne stocke que le contenu qui l'intéresse, plus quelques informations d'indexation qui aident à déterminer quel nœud stocke quoi.

IPFS stocke un fichier et crée une empreinte unique appelée hachage cryptographique. En effet, les fichiers avec leur contenu haché sont adressés dans une table de hachage distribuée (DHT) tout en supprimant les fichiers doublons sur le réseau [74]. Cette table de hachage permet d'assurer un accès rapide aux fichiers en retournant la liste des nœuds qui stockent le contenu demandé. La figure suivante montre un exemple de fonctionnement d'IPFS.

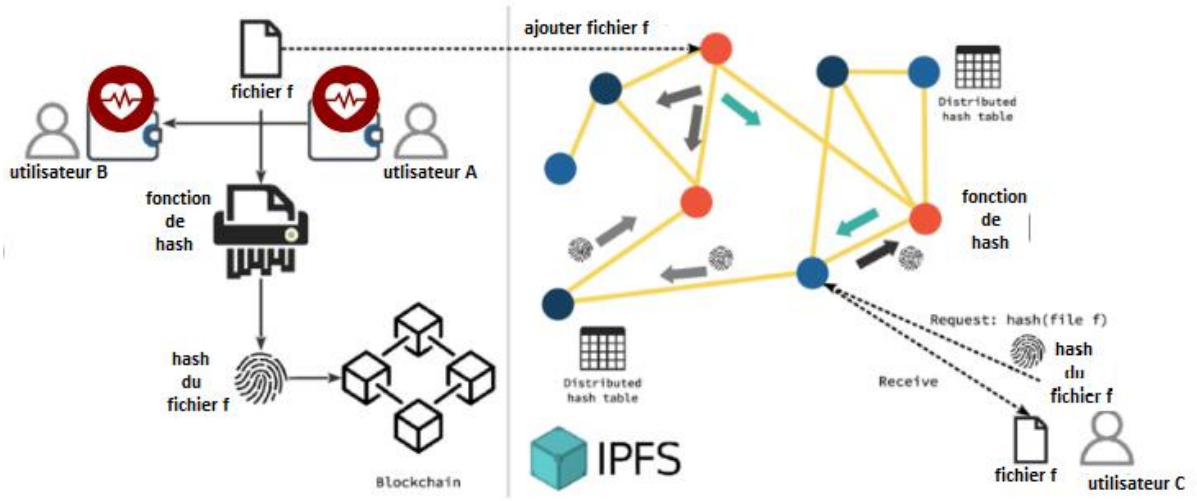

Figure 16 : fonctionnement de l'IPFS avec blockchain [75]

Dans notre solution, au lieu de stocker les données médicales complètes d'un patient dans le réseau blockchain, seul le hash du contenu des données va être stocké comme illustré dans la Figure 17.

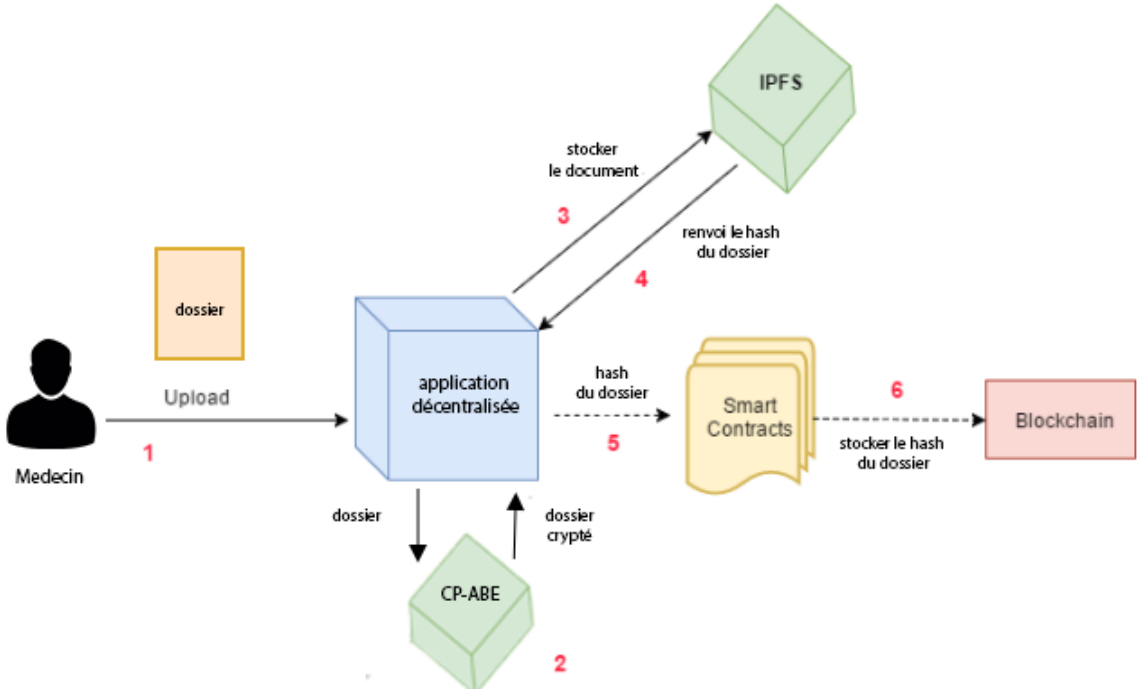

Figure 17: schéma de stockage d'un dossier dans notre application.[76]

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

Dans notre système, l'utilisateur possède un ou plusieurs attributs qui sont gérés par l'autorité de confiance. Le patient a un attribut de valeur dynamique qui diffère d'un patient à l'autre.

Le médecin a des attributs qui dépendent de leurs spécialités et domaine de travail, les attributs. La liste des attributs est définie comme étant : { cardiologie , chirurgie , gastro-entérologie , dermatologie , gynécologie , médecine générale , neurologie , ophtalmologie , psychiatrie , radiologie , rhumatologie}. Cette liste peut être modifiée.

Notre solution 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 patients définissent des politiques d'accès pour le chiffrement. Seuls les médecins 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 :

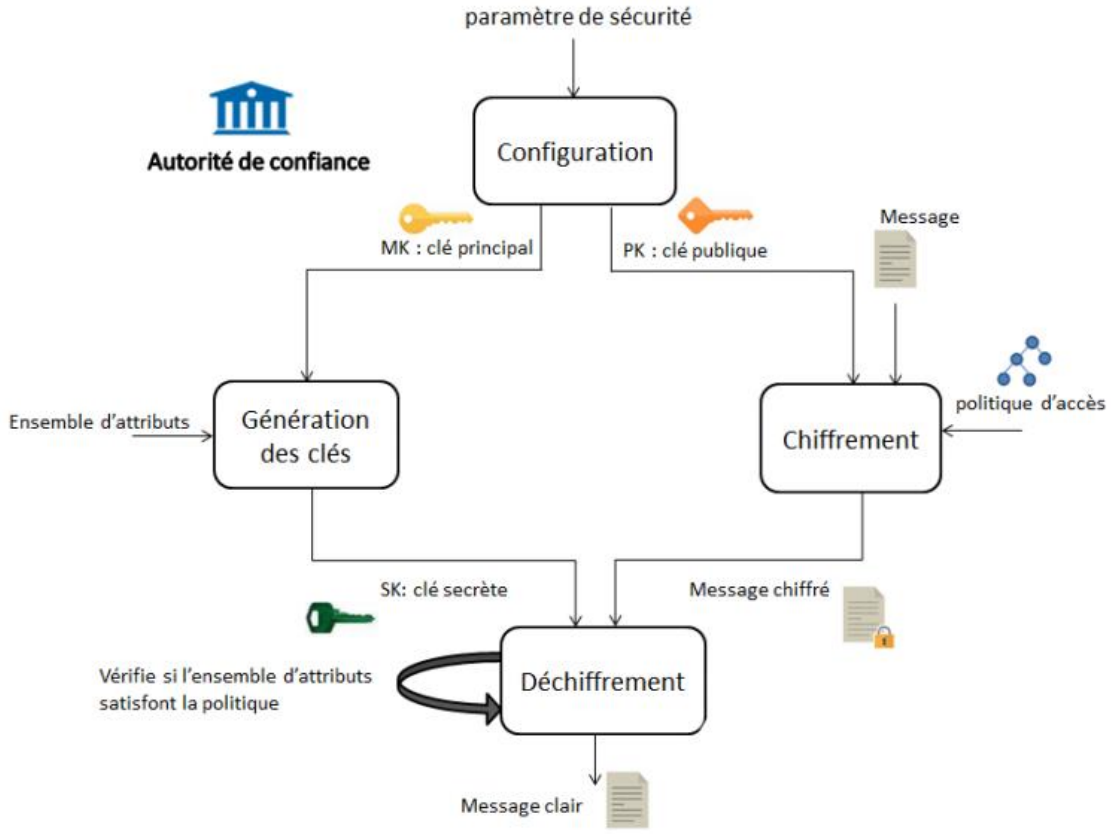

Figure 18: algorithme CP-ABE

1.4 Fonctions de Hachage

Le hachage à un rôle très important dans notre application car il est utilisé au niveau de blockchain et au niveau du Cloud IPFS :

- Comme mentionné dans le chapitre 2, blockchain est une technologie à base cryptographique où le hachage est utilisé pour assurer la validité et cohérence de notre blockchain vu que chaque bloc garde le hash du bloc précèdent. La fonction de hachage utilisée est Keccak 256(SHA3) [75].
- Le cloud IPFS sauvegarde le hash des documents. La fonction de hachage utilisée est SHA-256 [78].

1.4.1. Le Hachage SHA-256

SHA-2 (Secure Hash Algorithm) est une famille de fonctions de hachage qui ont été conçues par la National Security Agency des États-Unis (NSA), sur le modèle des fonctions SHA-1 et SHA-0. Telle que décrite par le National Institute of Standards and Technology (NIST), elle comporte les fonctions, SHA-256 et SHA-512 dont les algorithmes sont similaires mais opèrent

sur des tailles de mot différentes (32 bits pour SHA-256 et 64 bits pour SHA-512) expliquer dans la figure suivante :

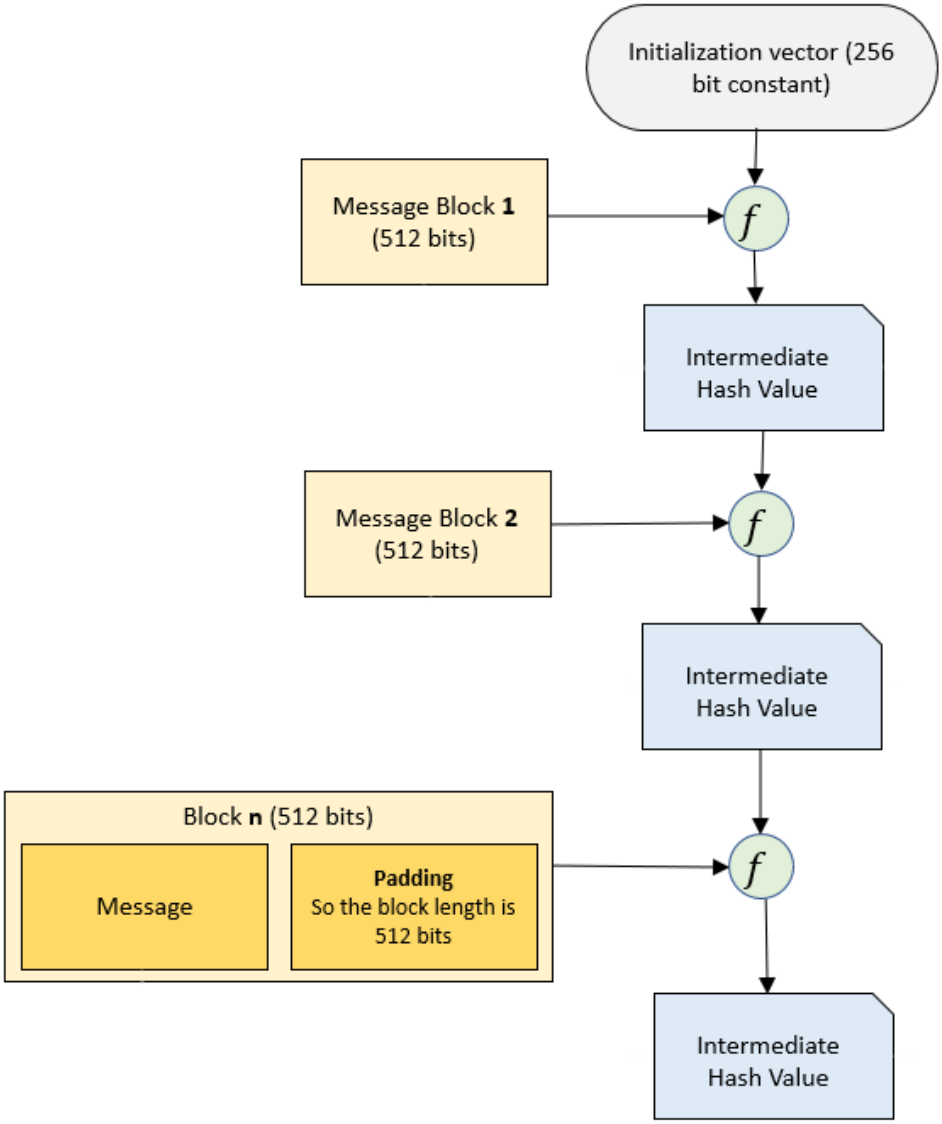

Figure 19:Algorithme SHA-256[78]

À un niveau élevé, l'algorithme SHA-256 fonctionne comme suit :

1. 1. Prendre le message d'entrée et s'assurer que sa longueur (en bits) est un multiple de 512 bits. Cela se fait en ajoutant un remplissage.
2. Prendre le message passé et diviser le en blocs N de taille 512 bits.
3. Pour chaque bloc, calculer sa valeur de hachage :intermédiaire :

a. Initialiser le calendrier des messages, une séquence de 64 mots de 32 bits
b. Initialiser les huit variables de travail a ... h avec les valeurs de hachage H0 ... H7 de l'itération précédente (pour la première itération, H0 ... H7 sont initialisés avec des constantes).

c. Effectuer 64 itérations où les variables de travail a ... h sont tournées d'une certaine manière. C'est le cœur de la fonction de hachage. Dans cette étape, le message est inséré dans le hachage en utilisant beaucoup de mélange bitwise.
d. Calculer les nouvelles valeurs de hachage intermédiaires H0 ... H7 comme H0 = H0 + a, H1 = H1 + b et ainsi de suite.

4. Concaténer H0 ... H7 comme condensé du message et le retourner.

Plus de détail sur chaque étape est donné dans [78].

1.4.2. Hachage Keccak 256(SHA-3)

La fonction de hachage de Keccak est basée sur le standard SHA-3 qui est différent de celui de SHA-1 et SHA-2. L'idée clé derrière SHA-3 est basée sur des permutations non clé, par opposition à d'autres constructions de fonctions de hachage typiques qui utilisaient des permutations clé. Keccak n'utilise pas non plus la transformation Merkle-Damgard qui est couramment utilisée pour gérer les messages d'entrée de longueur arbitraire dans les fonctions de hachage. Une nouvelle approche appelée construction d'éponge et de pression est utilisée dans Keccak (Figure 20). C'est un modèle de permutation aléatoire. Le détails d'implémentation est donnée dans [79].

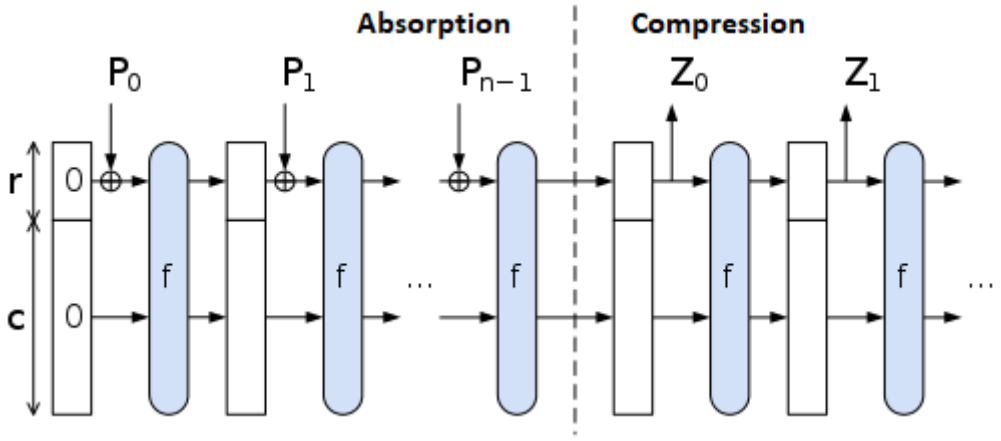

Figure 20: Construction d'éponge et de pression[79].

2 Architecture de notre application :

Dans l'architecture de notre application (figure 21), nous distinguons cinq composants (éléments) :

- **Autorité de confiance :** qui est responsable de la génération des clés publique, principale et secrètes que nous utiliserons pour le chiffrement et déchiffrement du CP-ABE. Elle est chargée aussi de l'émission des attributs pour les utilisateurs de notre application pour générer les clés secrète de ses derniers.
- **Patient :** qui crée l'accès pour ses dossiers pour les médecins, et consulte son dossier.

- **Médecin :** qui ajoute les fiches de suivi dans le dossier médical d'un patient si ses attributs sont conformes à la politique d'accès défini par le patient.
- **Réseau Blockchain :** qui permet de nous assurer l'immutabilité et l'intégrité grâce au consensus qui permet aux nœuds de notre blockchain de vérifier la validité de l'information avant de sauvegarder les données. Ces données peuvent être de plusieurs types : les informations d'authentification, l'emplacement des données dans le cloud, le hash du dossier médical stocké dans le cloud et les attributs nécessaires pour le contrôle d'accès et le chiffrement CP-ABE.
- **Serveur Cloud IPFS :** qui sert à stocker les dossiers médicaux chiffrées par l'algorithme CP-ABE et crée un hash de ce dernier qui sera stocké avec l'emplacement du dossier dans notre blockchain pour avoir accès au dossier par la suite.
- **Application décentralisée :** qui aide et fait la relation entre les différentes parties de notre réseau distribué.

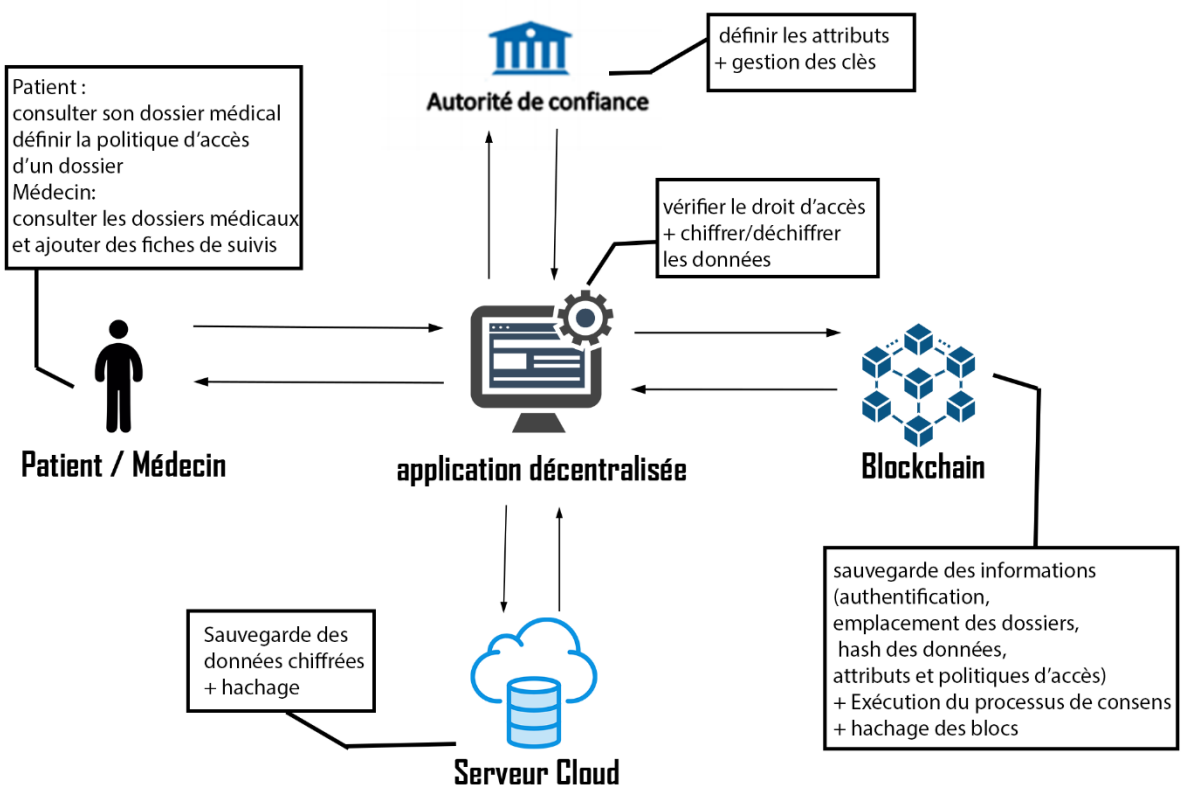

Figure 21: architecture de l'application

Notre plateforme permet de gérer les données médicales notamment la consultation et la modification :

- La modification se fait par le médecin en ajoutant une fiche de suivi au dossier. Elle déclenche le processus de sauvegarde.
- La consultation se fait par le patient ou le médecin et elle déclenche le processus du téléchargement.

Ces deux processus seront détaillés par la suite.

2.1. Processus de modification

Dans ***la figure 22,*** nous expliquons l'ajout de nouvelle fiche de suivi par le médecin dans le dossier de patient :

1. Le médecin demande d'ajouter un fichier ou une fiche de suivi
2. L'application décentralisée envoi une demande de la clé publique
3. L'autorité envoie la clé publique
4. L'application crypte les données par l'algorithme CP-ABE et envoi les données cryptées au cloud. Notons ici que le chiffrement réussi seulement dans le cas om le médecin possède le droit d'accès au dossier selon la politique d'accès définie par le patient, sinon une erreur sera affichée.
5. Le cloud sauvegarde les données et envoi l'emplacement du fichier (des données) avec un hash du fichier.
6. L'application stocke l'emplacement et le hash du fichier dans un nœud de blockchain qui déclenche le processus de consensus PoW entre les nœuds.

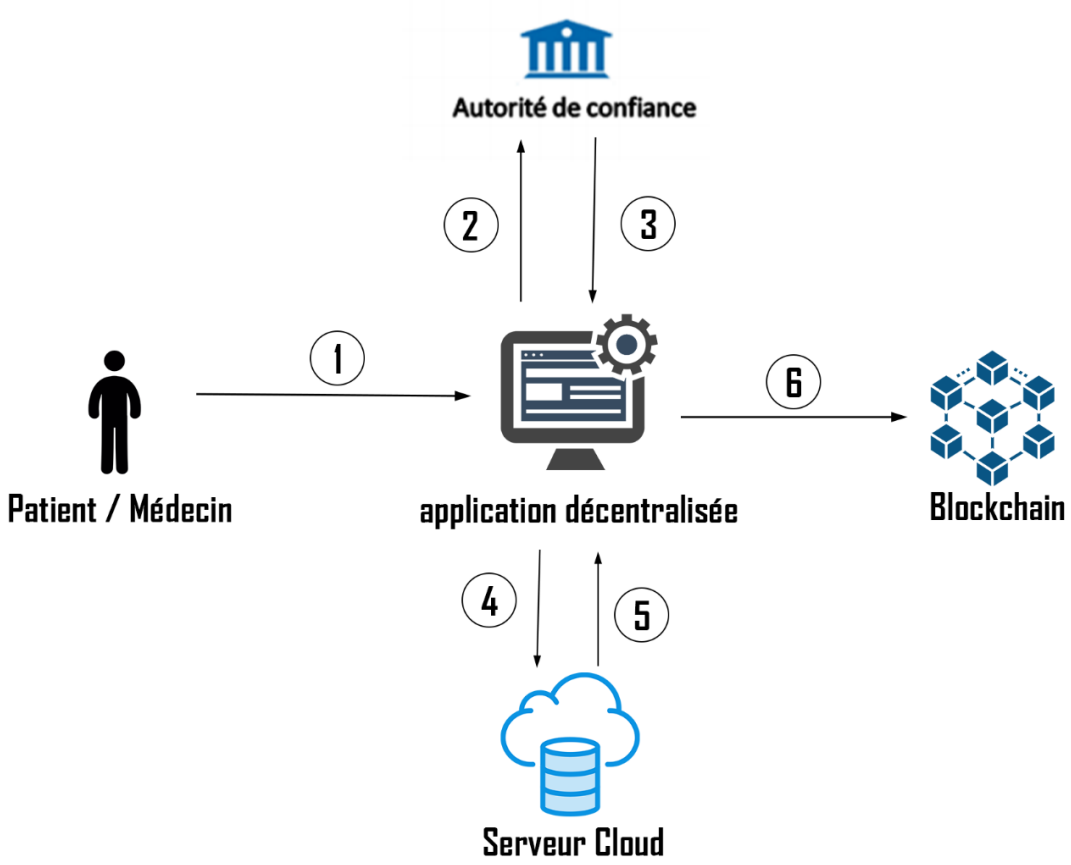

Figure 22: stockage d'une fiche.

2.2. Processus de consultation

Dans ***la Figure 23,*** nous allons expliquer comment les données circulent dans notre application lors de la requête de lecture du dossier

1. L'utilisateur demande l'accès à un dossier
2. L'application demande à un nœud de blockchain les attributs de l'utilisateur et la politique d'accès au dossier ainsi que l'emplacement du dossier crypté stocké dans le cloud.
3. Le nœud répond à la requête de l'application
4. L'application envoi les attributs de l'utilisateur pour générer la clé sécrète
5. L'autorité de confiance envoi la clé secrète
6. Application demande le dossier au cloud
7. Le Cloud répond par le dossier demandé qui est crypté.
8. L'application déchiffre le dossier avec la clé secrète et l'envoi à utilisateur. Notons ici que le déchiffrement réussi seulement dans le cas où l'utilisateur possède le droit d'accès au dossier, sinon une erreur sera affichée.

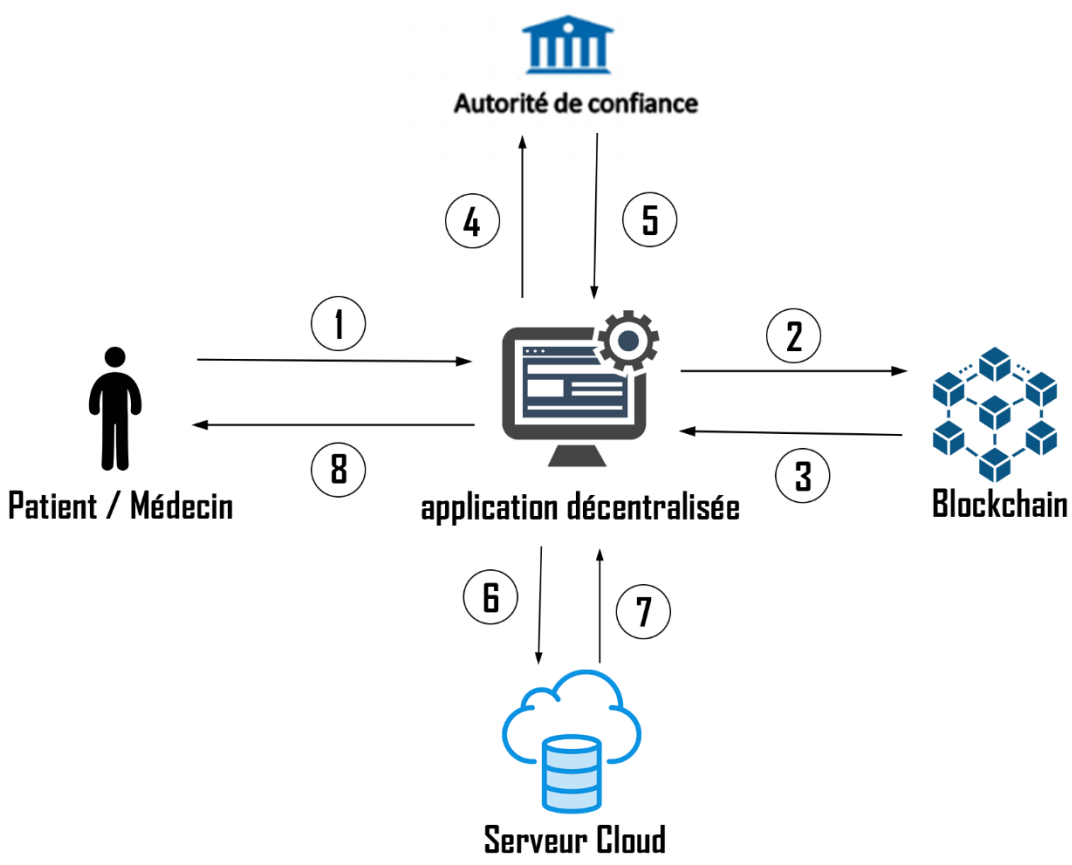

Figure 23 : téléchargement d'un dossier

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

3.1. Diagramme de cas d'utilisation :

Un diagramme de cas d'utilisation est utilisé pour représenter les besoins des utilisateurs par rapport au système utilisé. Dans notre application, nous avons les acteurs suivants :

- Médecin : l'individu qui peut accéder aux dossiers médicaux des patients
- Patient : l'individu qui peut accéder à son dossier médical
- Autorité de confiance : l'entité responsable de la gestion des attributs des utilisateurs et les clés.

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

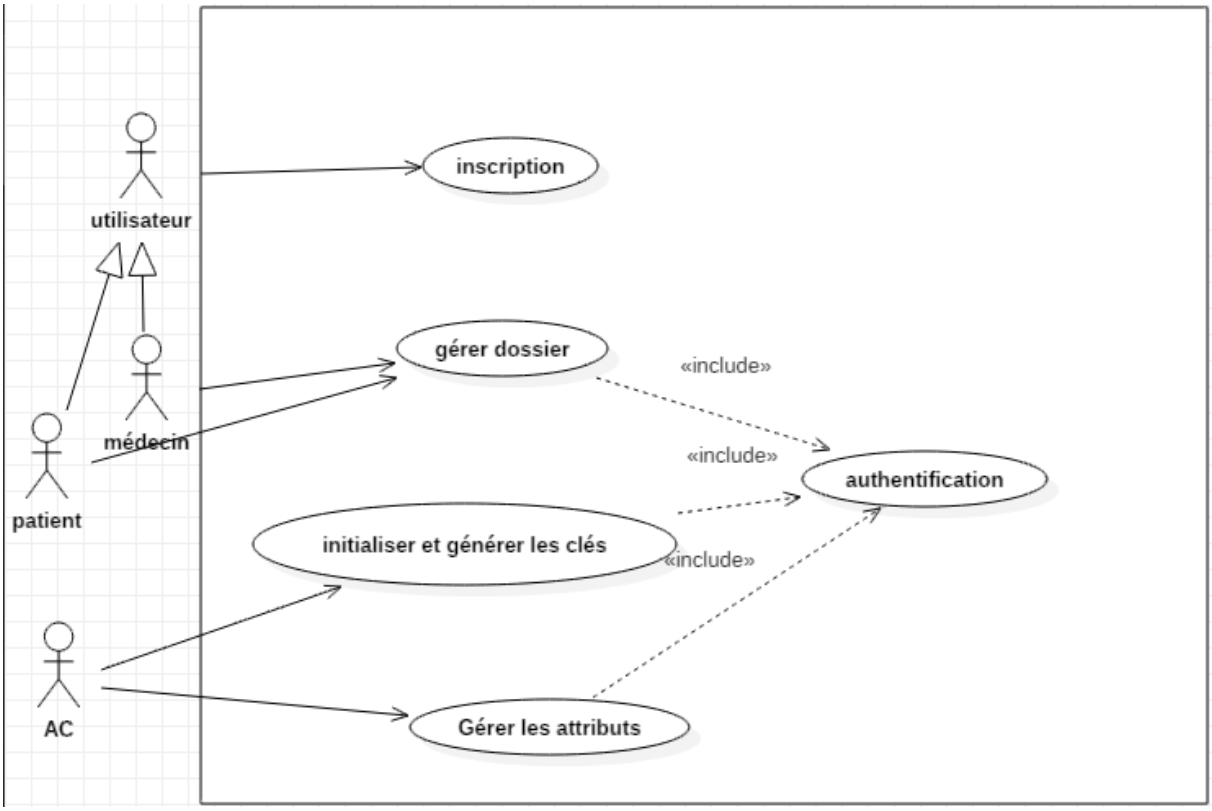

Figure 24: diagramme de cas d'utilisation globale

Cas d'utilisation	Acteurs	Description
Inscription	utilisateur	Les acteurs peuvent s'inscrire sur l'application
Authentification	Patient/medecin AC	Doivent s'authentifier pour accéder à l'application
Gérer dossier médical (Figure 25)	Patient/Médecin	Les acteurs peuvent rechercher, consulter les dossiers médicaux. Les médecins peuvent ajouter des fiches de suivis Le patient peut créer des politiques d'accès pour son dossier
Générer les clés (Figure 26)	Autorité de Confiance (AC)	Générer les clés de chiffrement et de déchiffrement et les envoie aux parties concernées.
Gérer les attributs		Définir les attributs des acteurs interagisse dans notre system

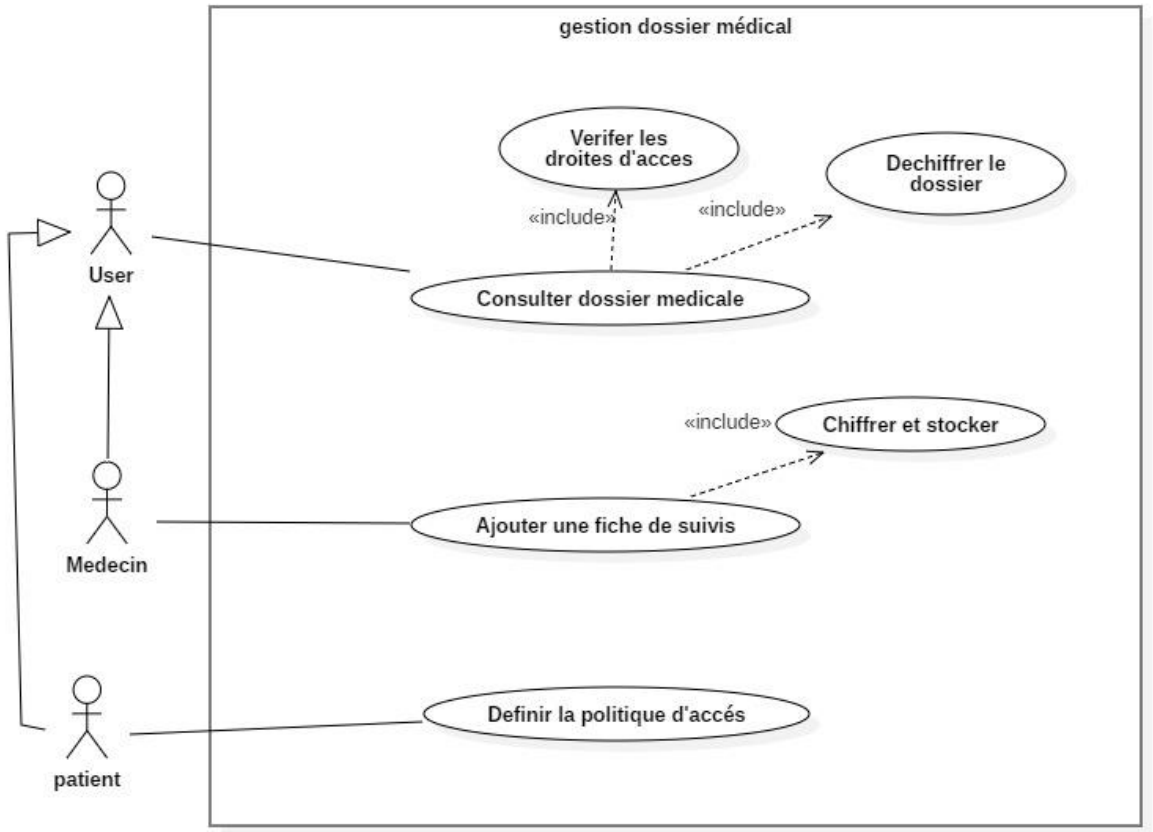

Figure 25 : Diagramme de cas d'utilisation gestion du dossier médical

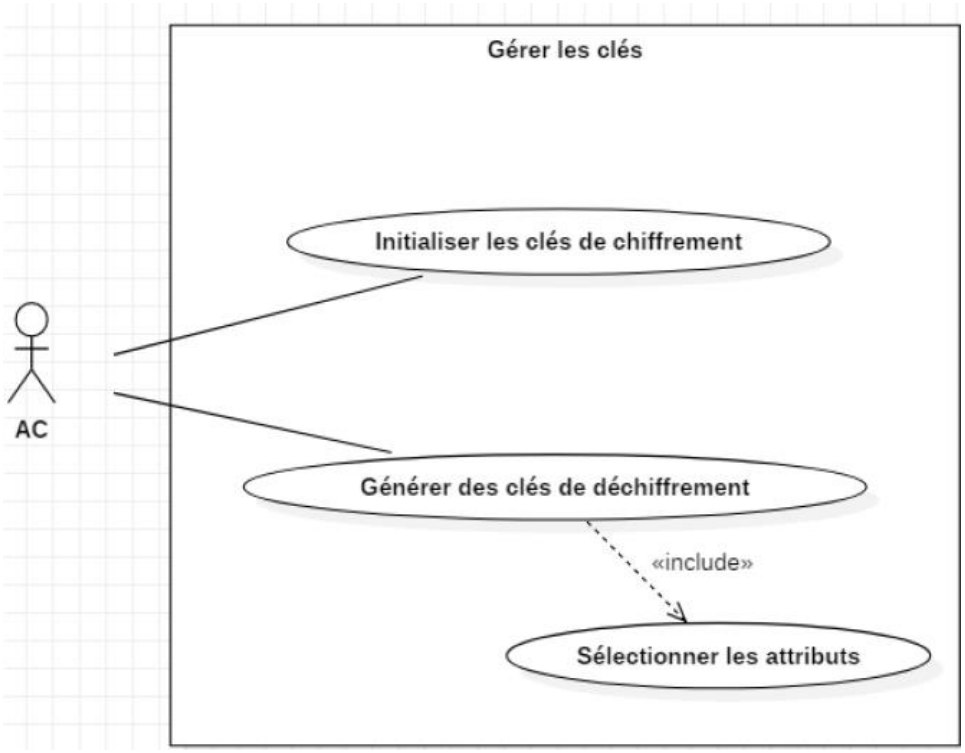

Figure 26 :Diagramme de cas d'utilisation gestion des clés

3.2 Diagrammes de séquence :

Dans cette section, nous allons essayer d'expliquer le fonctionnement de quelques fonctions principales de notre application : Inscription d'un utilisateur, Authentification d'un utilisateur, Gestion des attributs des utilisateurs, Chiffrement et Stockage et Téléchargement et déchiffrement.

3.2.1 Inscription d'un utilisateur :

Les utilisateurs (patient/médecin) doivent être inscrits dans notre application pour pouvoir interagir avec cette dernière. Pour cela, ils devront remplir un formulaire avec les informations suivantes : Numéro de sécurité sociale, nom, prénom, date de naissance.

Le diagramme de séquence suivant décrit les étapes nécessaires pour l'inscription d'un utilisateur :

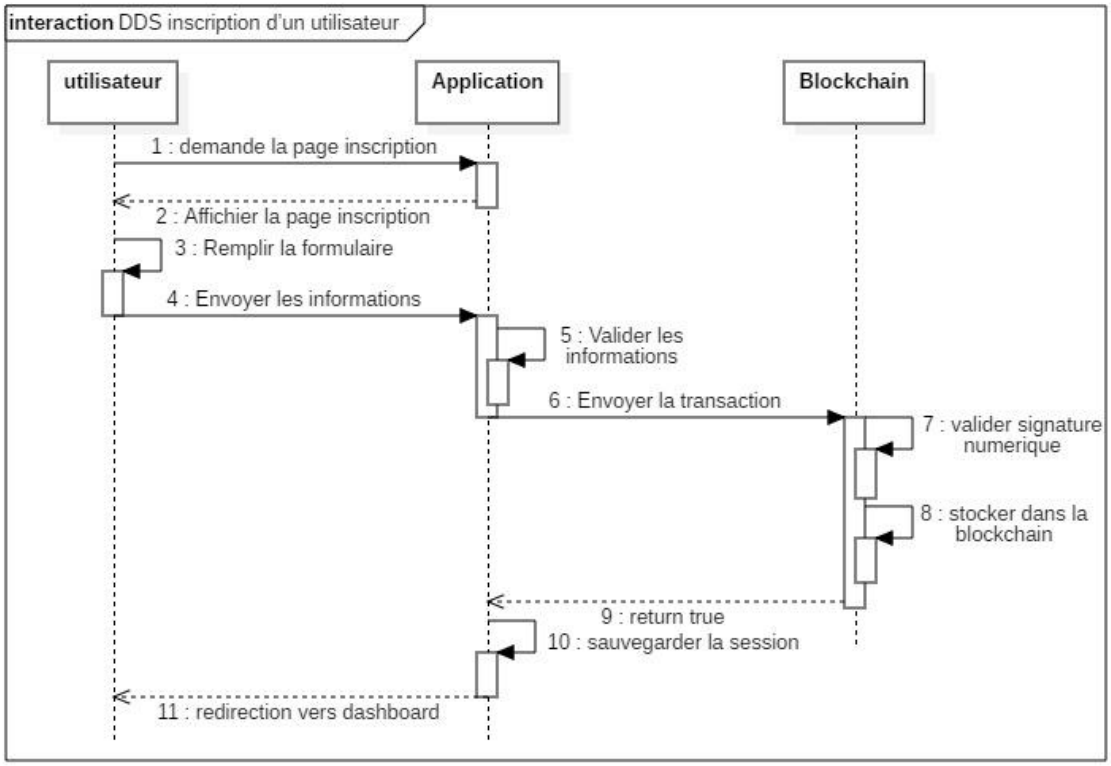

Figure 27: Diagrammes de séquence inscription d'un utilisateur

3.2.2 Authentification d'un utilisateur :

Pour accéder à notre application et ses fonctionnalités, les utilisateurs doivent d'abord s'authentifier.

Le diagramme de séquence suivant explique les étapes nécessaires pour l'authentification :

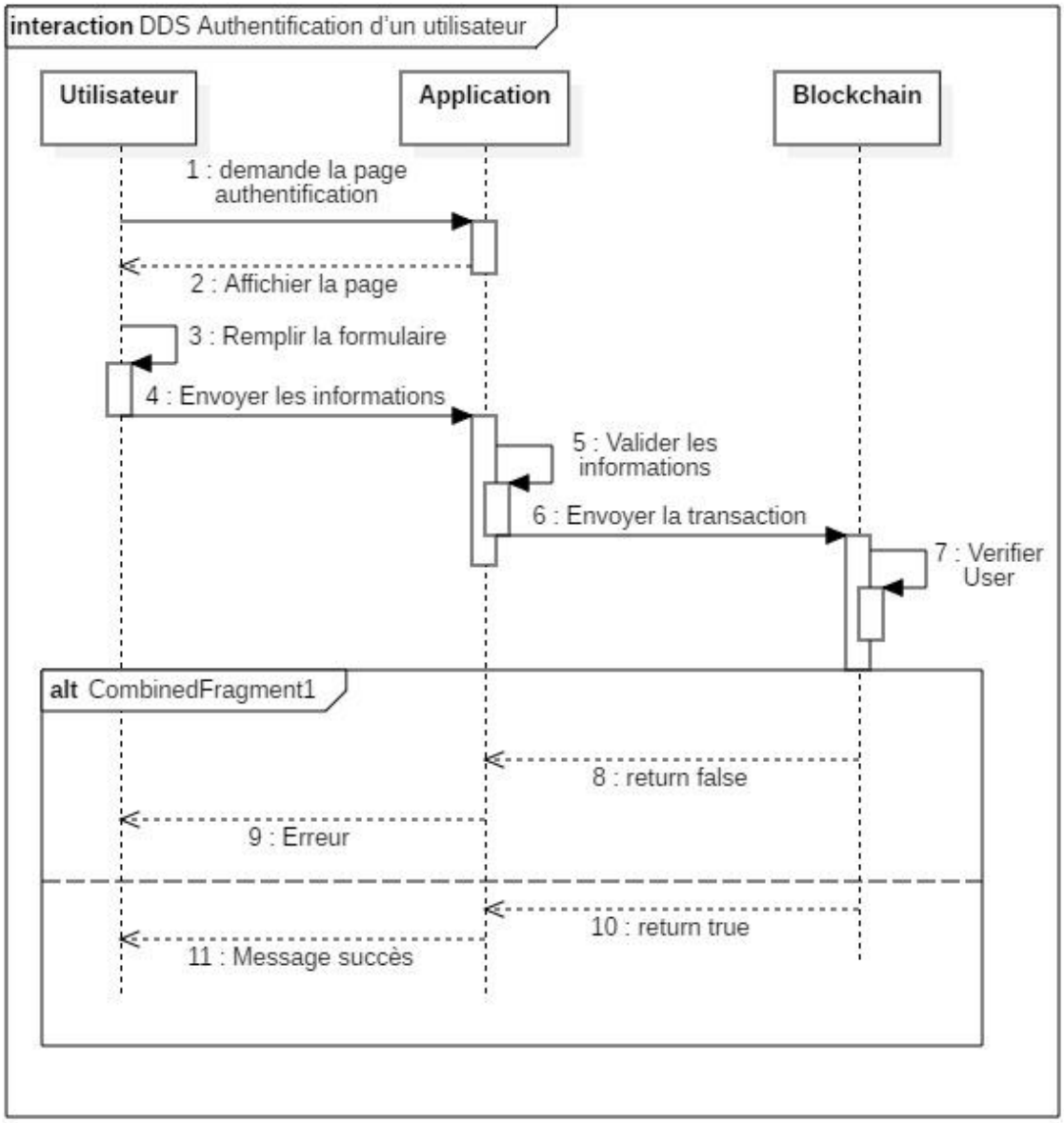

Figure 28 : Diagrammes de séquence authentification d'un utilisateur

3.2.3 Gestion des attributs des utilisateurs :

Dans notre application, nous avons utilisé un contrôle d'accès à base d'attribut. L'autorité de confiance est responsable de la distribution des attributs à la personne concernée. Les étapes nécessaires pour réaliser cela sont décrites dans le diagramme de séquence suivant :

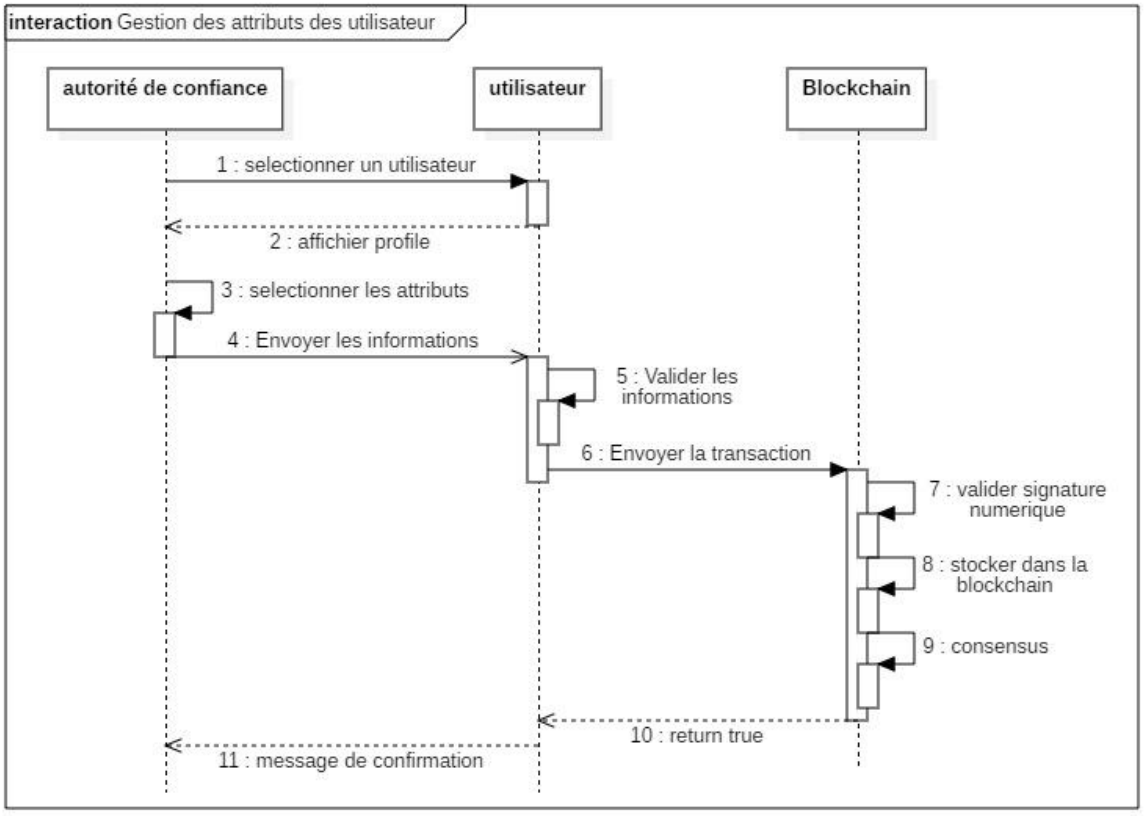

Figure 29: Diagrammes de séquence Gestions des attributs des utilisateurs

3.2.4 Chiffrement et Stockage :

Pour des raisons de sécurité, les données médicales doivent être chiffrées en utilisant la technique de chiffrement CP-ABE expliquée précédemment. Les étapes nécessaires pour réaliser cela sont décrites dans le diagramme de séquence suivant :

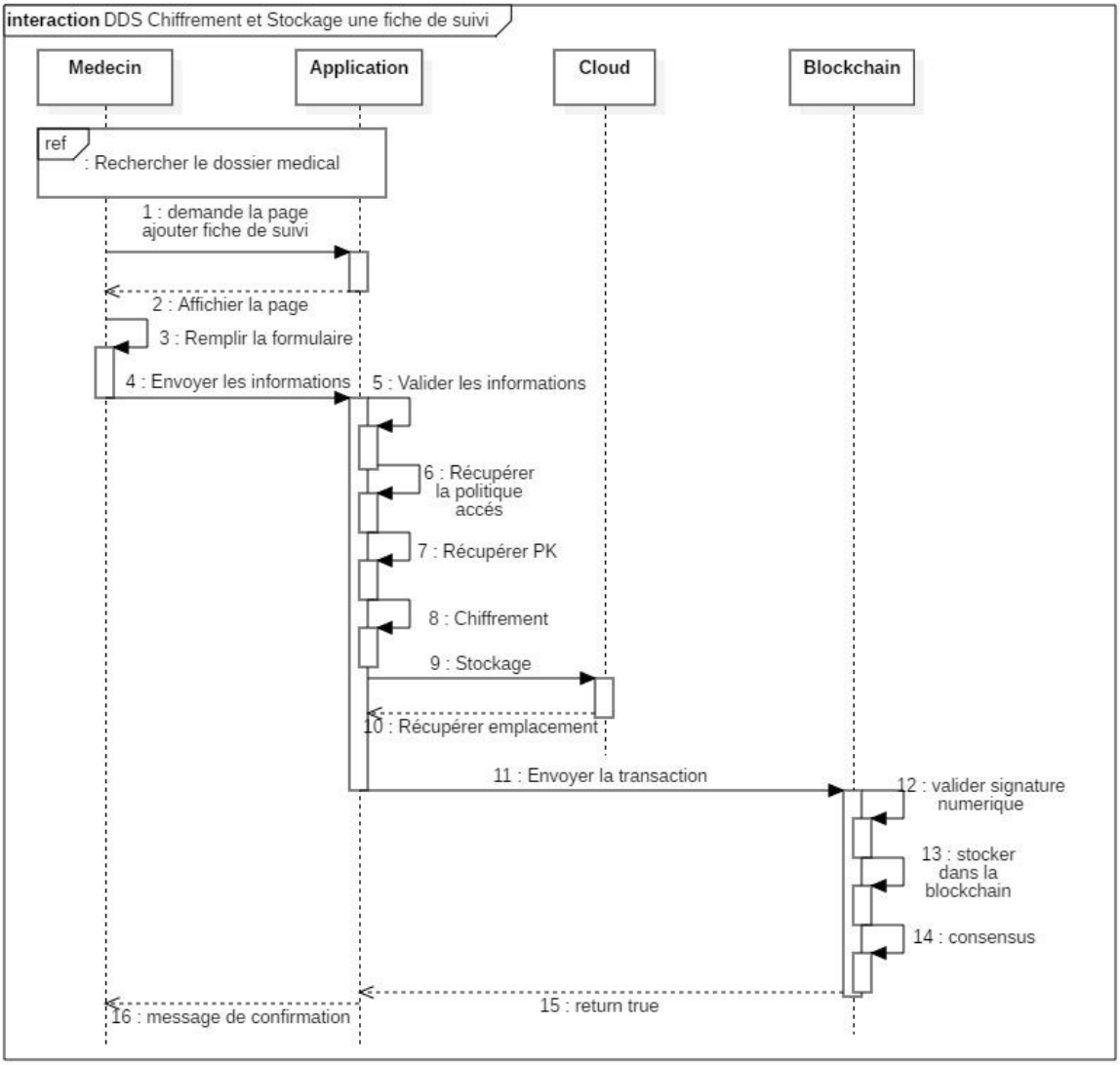

Figure 30 : Diagrammes de séquence chiffrement et stockage d'une fiche de suivi

3.2.5 Téléchargement et Déchiffrement :

Comme expliqué précédemment, les données médicales sont chiffrées puis stockées. Ainsi pour pouvoir récupérer ces dernières en claire, nous devons trouver le dossier et avoir les attributs correspondants pour déchiffrer et télécharger ce dernier. Les étapes nécessaires pour réaliser cela sont décrites dans le diagramme de séquence suivant :

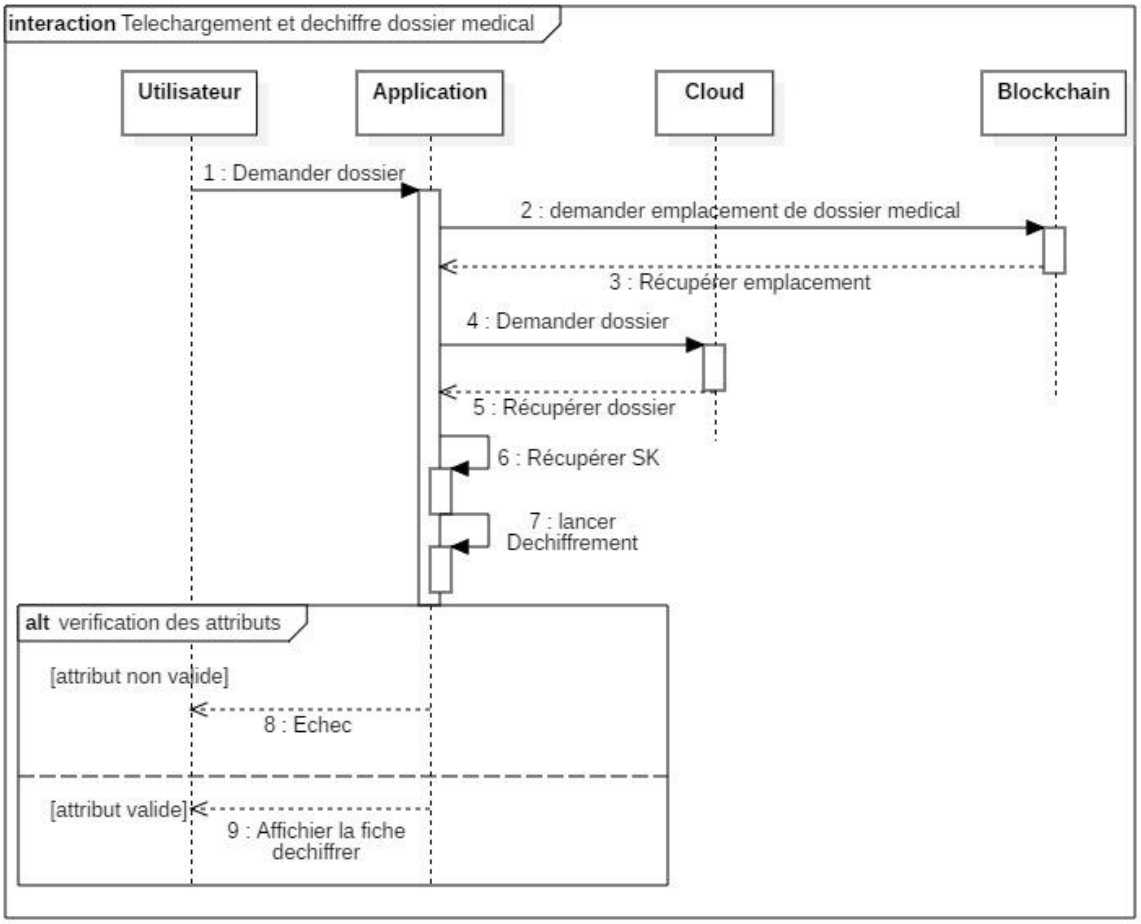

Figure 31 : Diagrammes de séquence téléchargement et déchiffrement d'un dossier médical

4 Conclusion :

Dans ce chapitre, nous avons décrit notre solution pour sécuriser aux mieux notre application e-santé. L'approche utilisée repose sur l'utilisation (i) de la technologie blockchain privée qui garantit l'immutabilité et la traçabilité (ii) un cloud (IPFS) qui assure la disponibilité des données, et (iii) avec l'algorithme de chiffrement par attribut CP-ABE qui assure le contrôle d'accès, la confidentialité et l'intégrité des données..

Dans le chapitre suivant, nous expliquerons les étapes suivis pour l'implémentation et la réalisation de cette plateforme d'E-santé.

Chapitre 4 : développement de la solution

Dans ce chapitre, nous allons parler de la 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 les outils logiciels utilisés pour développer notre plateforme. Ensuite, nous décrivons l'implémentation des approches proposées (contrôle d'accès basé sur le chiffrement CP-ABE, sauvegarde des données dans le blockchain et le cloud). Enfin, nous présenterons les interfaces graphiques de notre application**.**

1 Environnement de développement :

La **figure 32** montre l'interaction basique entre les différentes technologies utilisées dans notre plateforme.

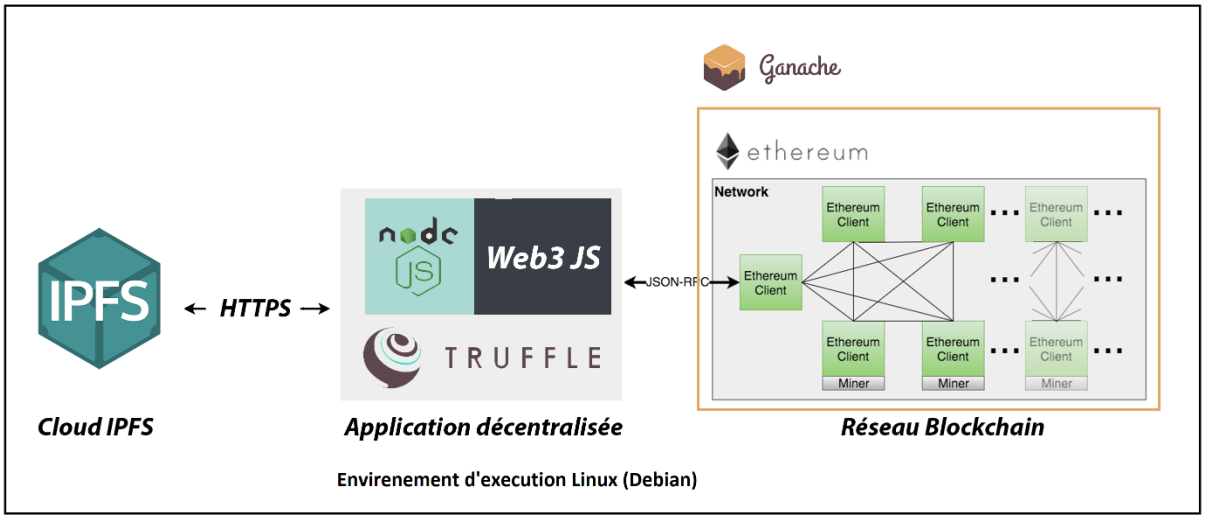

Figure 32: technologies utilisées dans notre projet

- **Coté Cloud**, nous avons installé le package IPFS « dist.ipfs.io »[2]
  Son installation nécessite la configuration de la clé pour créer et lancer un réseau privé

[2] https://dist.ipfs.io/go-ipfs/v0.9.0/go-ipfs_v0.9.0_linux-amd64.tar.gz

(figure 33). Pour rendre IPFS un cloud, nous avons généré un fichier avec nos propres clés en utilisant l'outil swarmkey[3].

```
1989  wget https://dist.ipfs.io/go-ipfs/v0.9.0/go-ipfs_v0.9.0_linux-amd64.tar.gz
1990  tar -xvzf go-ipfs_v0.9.0_linux-amd64.tar.gz
1991  cd go-ipfs
1992  sudo bash install.sh
1993  ipfs --version
1994  history
kali@kali:~/go-ipfs$ ipfs init
generating ED25519 keypair...done
peer identity: 12D3KooWLvg3TPPukSjhxG2uxGRuAsKhnmmzGAqbNBEVu8PKShiK
```

Figure 33: Instalation ipfs

- **Coté Blockchain,** nous avons opté pour Ethereum (Figure 34) que nous allons décrire en détails dans la section suivante.
- **Coté application décentralisée,** nous avons utilisé DAPPS (application décentralisé) Web3 JS [77] qui permet à n'importe qui de participer sans monétiser ses données personnelles.[80]. La figure 34 montre l'interaction entre application Web3jS et le blockchain Ethereum. web3.js fournit une abstraction de l'interface Ethereum json-rpc qui permet d'interagir avec un nœud Ethereum à l'aide de JavaScript brut. En termes simples, web3.js expose les API JSON-RPC en tant qu'API JavaScript. C'est pourquoi, il était nécessaire d'utiliser de plus ces deux outils :
    - Node JS [81] qui est frame work Javascript que nous allons utiliser pour développer notre application sur le serveur, responsable de l'exécution des commande de Web3.JS, Truflle et CPABE et pour héberger notre application sur les nœud [82].
    - Framework Truffle [83] qui fournit une suite d'outils pour développer des contacts intelligents Ethereum avec le langage de programmation Solidity [80]. Nous revenons sur cet outil dans la section suivante.

[3] commande de téléchargement : go get -u github.com/Kubuxu/go-ipfs-swarm-key-gen/ipfs-swarm-key-gen. commande d'exécution :./go/bin/ipfs-swarm-key-gen > ~/.ipfs/swarm.key

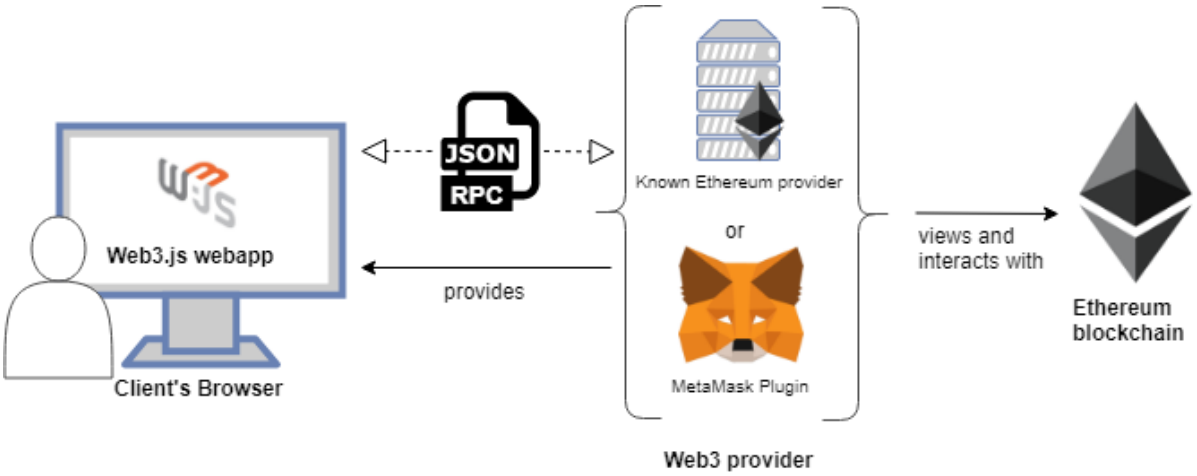

Figure 34: Interaction de Web3.js et Ethereum

2 Blockchain Ethereum :

Dans l'univers d'Ethereum, il y a un seul ordinateur canonique (appelé la machine virtuelle d'Ethereum, ou EVM) dont tout le monde sur le réseau d'Ethereum est d'accord sur son état (figure 35). Quiconqu'il participe au réseau Ethereum, chaque nœud Ethereum conserve une copie de l'état de cet ordinateur. De plus, tout participant peut diffuser une demande pour que cet ordinateur effectue des calculs arbitraires. Chaque fois qu'une telle demande est diffusée, d'autres participants sur le réseau vérifient, valident et effectuent (« exécutent ») le calcul. Cela provoque un changement d'état dans l'EVM, qui est engagé et propagé dans tout le réseau. [80]

Les demandes de calcul sont appelées demandes de transaction ; l'enregistrement de toutes les transactions ainsi que l'état actuel de l'EVM est stocké dans le blockchain, qui à son tour est stocké et agrée par tous les nœuds. [80]

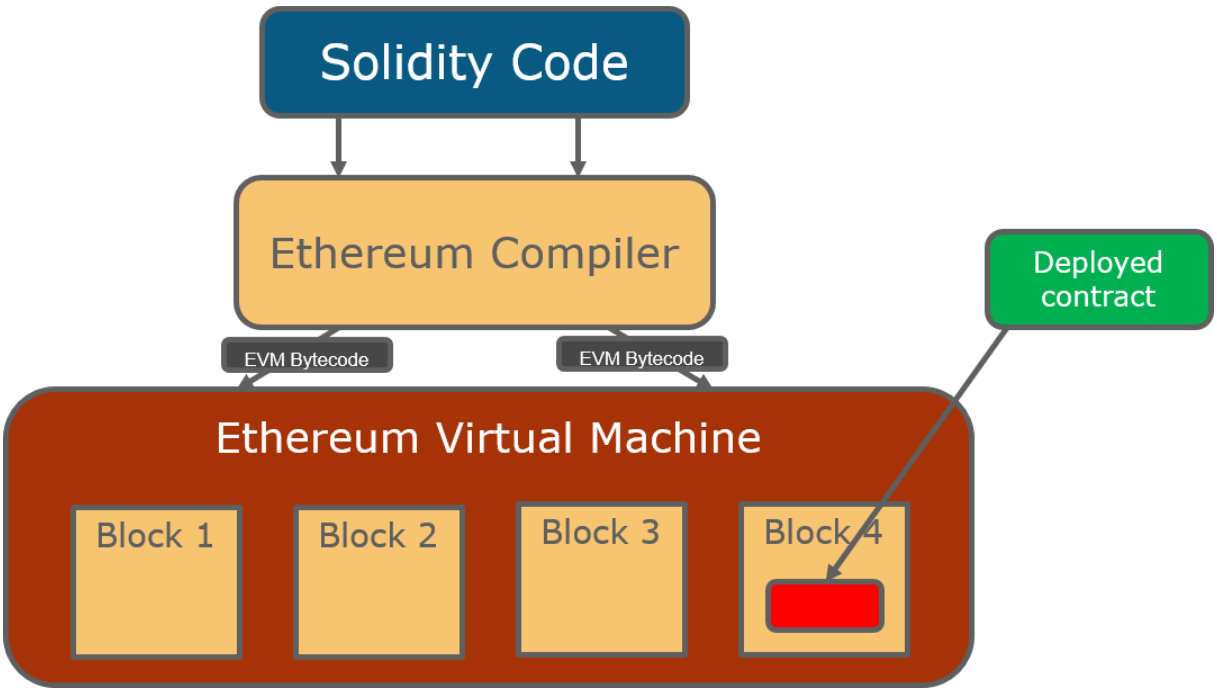

Figure 35: Ethereum Virtuel machine

2.1 Les composants principaux d'Ethereum

La figure 36, illustre la structure de notre utilisation de Ethereum

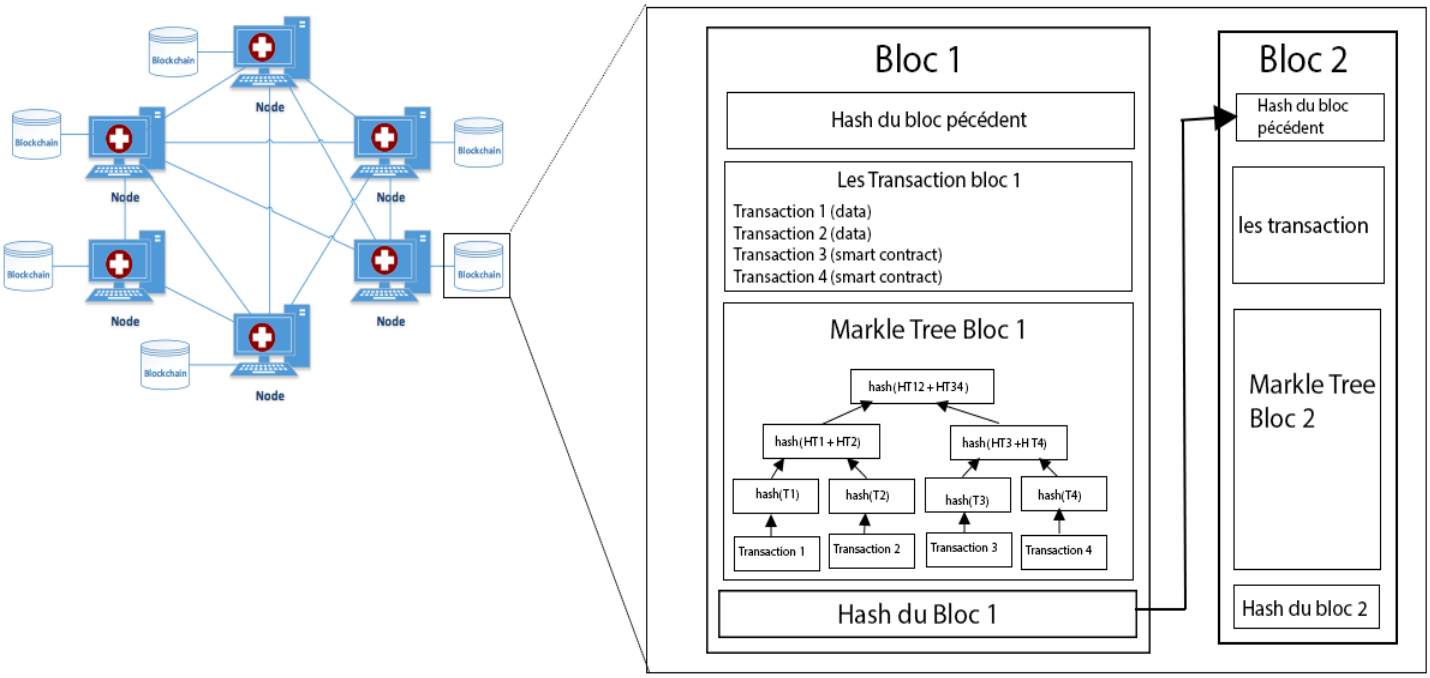

Figure 36 : Composants principal ethereum

- Une **transaction** dans Ethereum est un paquet de données qui contient des instructions, elle peut être une requête sur information existante dans le blockchain ou pour ajouter de nouvelles informations sur ce dernier ou la création d'un nouveau contrat dans le réseau qui pourra être exécutée.
- Un **arbre Merkle** (markerl tree) est une arborescence basée sur le hachage utilisé dans les systèmes distribués pour une vérification efficace sur l'intégrité d'un bloc. Chaque transaction est hachée en utilisant une fonction de hachage Keccak 256. Ensuite, chaque paire de fils est hachée récursivement jusqu'à ce que nous atteignions la racine, qui est un hachage de tous les transaction du bloc [84].
- Les **blocs** sont les principaux éléments constitutifs d'un blockchain, il contient un ensemble de transactions et le hash du bloc précédent dans sa chaîne.
- Un **contrat intelligent** (smart contract) est une collection de code (ses fonctions) et de données (son état) qui réside à une adresse spécifique sur le blockchain Ethereum. Ce code est écrit en Solidity.
- Un **nœud** est un ordinateur relié au réseau blockchain. Dans notre plateforme qui utilise un programme relayant les transactions qu'il reçoit et traite en cas de l'écriture d'une information il prend en charger le lancement du consensus pour validation de la transaction avec les autres nœuds du réseau

2.2 Mise en place d'un réseau Ethereum

Pour créer un réseau blockchain ethreum, nous avons suivi ces étapes :

1. **Installer Go-ethereum(Geth)** qui est implémenté avec le langage golang pour l'exécution et le test du blockchain Ethereum sur le côté client :

- Commande de téléchargement : $ Sudo add-apt-repsitory -y ppa:Ethereum/Ethereum
- Commande d'installation : $ Sudo apt install ethereum

2. **Création d'un blockchain privée**, on exécute les commandes suivantes :

- $ geth --datadir privchain init genesis.json
- $ geth --identity "newEth" --rpc --rpcport 8545 --rpcaddr 0.0.0.0 --rpccorsdomain "*" --datadir "privchain" --port 30303 --rpcapi "db,eth,net,web3" --networkid 999 console

3. **Installer Truffle** qui fournit une suite d'outils pour développer les contacts intelligents Ethereum avec le langage de programmation Solidity.

- Commande d'installation : *$ Sudo npm install -g truflle*
- Commande de création d'un projet truffle : *$ truffle init*

4. **Lancer Ganache** qui un Blockchain Ethereum personnel utilisé pour tester des contrats intelligents. Nous allons l'utiliser pour déployer nos contrats, développer des applications, et exécuter des tests.

- Commande de téléchargement : npm install ganache-cli@latest -g
- Commande de lancement :ganache-cli

Une fois notre blockchain est mis en place, nous avons passé à la création des contrats intelligents.

2.3. Création des contrats intelligents :

Dans cette section, nous allons présenter certaines parties des smart contract que nous avons créées et utilisées dans notre application :

- Le **smart contract du Medecin** qui contient l'identificateur, l'email, le mot de pass du medecin, et la table de mapping (attributs_medecin) qui permet de récupérer les attributs d'un médecin à partir de son identificateur.

```
contract Medecin {

  struct medecin {
      uint id;
      string email;
      string password;
  }

  mapping(uint => string[]) attributs_medecin;
  mapping(uint => medecin) public list_id_medecin;
  mapping(string => medecin) public list_email_medecin;
```

Figure 37: structure et mapping du smart contract medecin

- Le **smart contract du dossier médical** : qui contient un identificateur, un nom du dossier défini par le patient, le hash du fichier chiffré qui est stocké dans le serveur IPFS et en fin la politique d'accès pour le fichier.

```
pragma solidity ^0.5.16;
pragma experimental ABIEncoderV2;

contract DossierMedicale {

    struct Dossier {
        uint id;
        stirng NomDossier;
        string hash_file;
        string policy;
    }

    mapping(uint /* Patient_id */ => Dossier[] ) public list_dossier_user;

    function CreerDossier(uint _NomDossier,string memory _hash_file,string memory _policy) public  returns(bool){
     uint id= list_dossier_user[_patient_id].push(Dossier(0,_NomDossier,_hash_file,_policy)) -1;
     list_dossier_user[_patient_id][id].id=id;
     return true;
    }
```

Figure 38: structure, mapping et création du smart contract dossier médical

Une fois crées, ces contrats doivent être compilés et sauvegardés pour pouvoir être utilisées ultérieurement comme le montre la figure suivante :

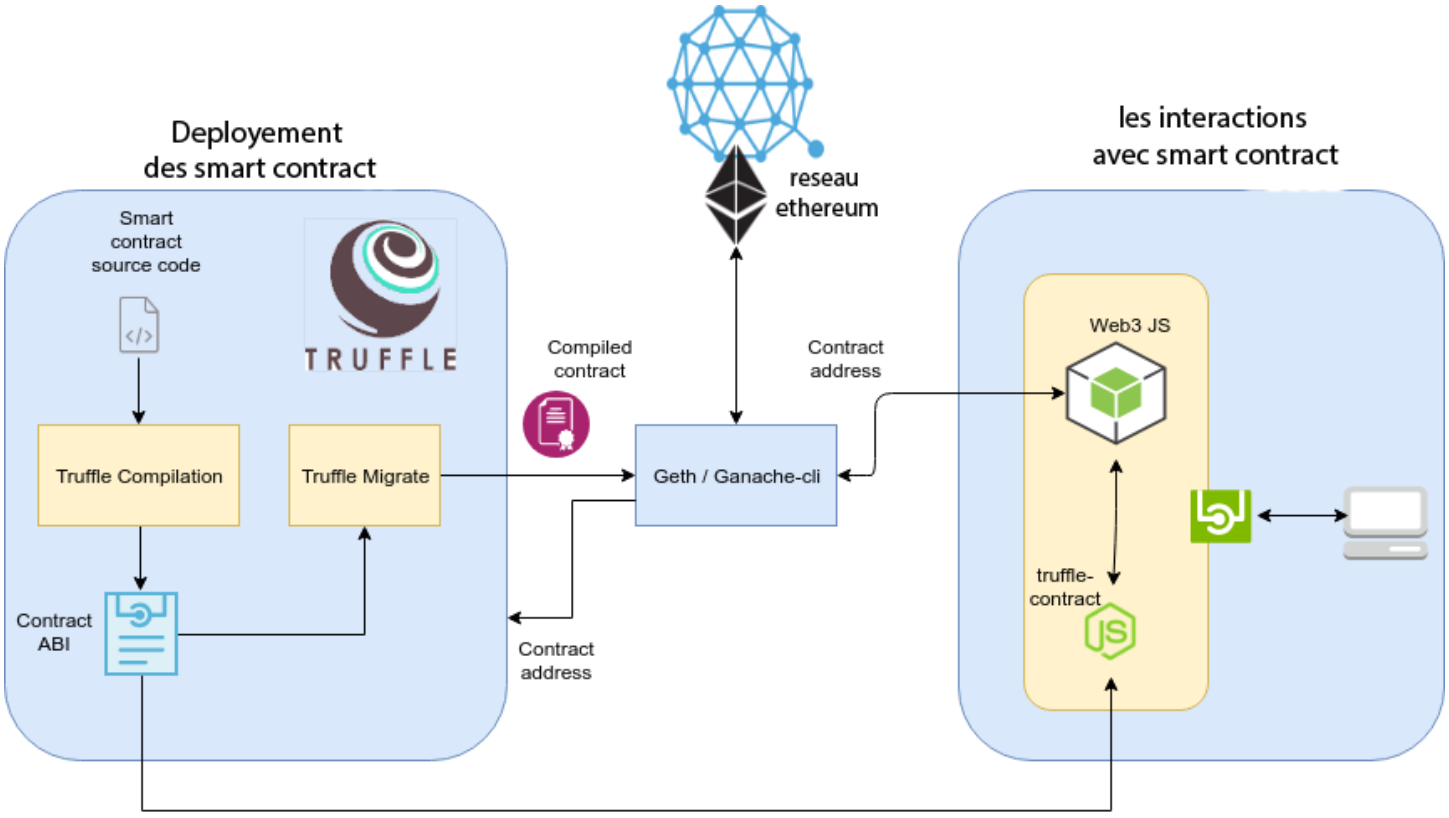

Figure 39: Contrats Intelligent

Voici l'interface de Ganache qui nous permet de déployer notre contrats :

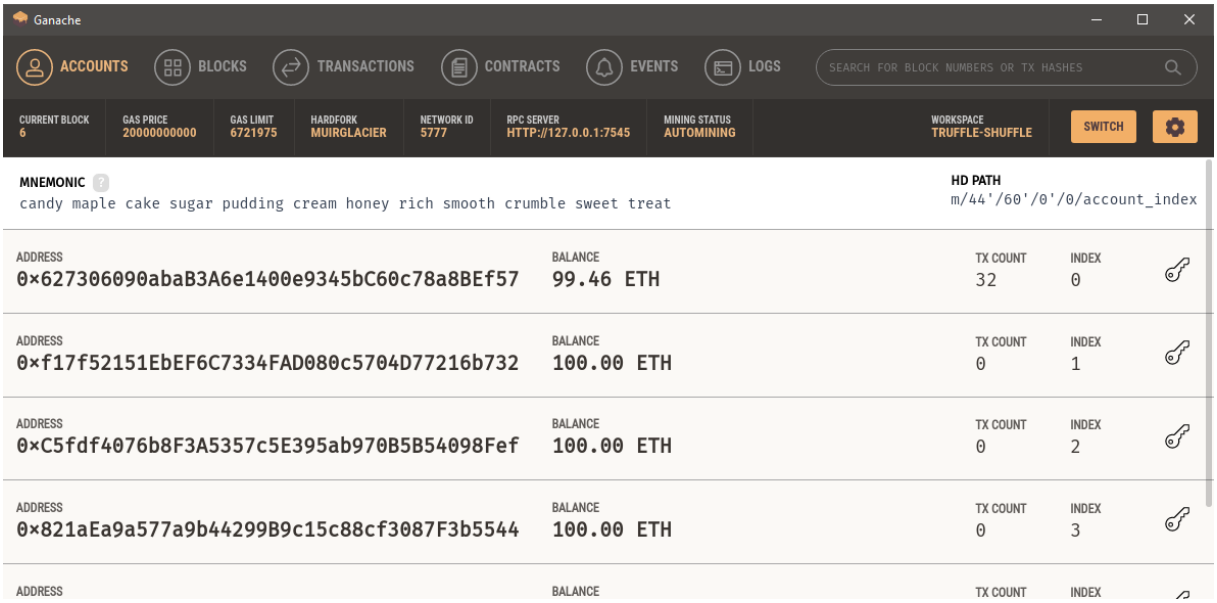

Figure 40: Interface ganache pour les contrats créés

2.4. Hachage des Transactions

Un arbre Merkle est une structure de données basée sur le hachage qui est une généralisation de la liste de hachage. C'est une arborescence dans laquelle chaque nœud de feuille est un hachage d'un bloc de données, et chaque nœud non transparent est un hachage de ses enfants. Typiquement, les arbres Merkle ont un facteur de ramification de 2, ce qui signifie que chaque nœud a jusqu'à 2 enfants.[84]

Les arbres Merkle sont utilisés dans les systèmes distribués pour une vérification efficace des données. Ils sont efficaces parce qu'ils utilisent des hachures au lieu de fichiers complets. Les hachages sont des moyens d'encodage de fichiers qui sont beaucoup plus petits que le fichier lui-même. Actuellement, leurs principales utilisations sont dans les réseaux peer-to-peer tels que Blockchain.

Les arbres Merkle sont généralement implémentés en tant qu'arbres binaires, comme indiqué dans la figure suivante : une donnée en entrée est divisée en blocs étiquetés (L1, …, L4). Chacun de ces blocs sont hachés en utilisant une fonction de hachage. Ensuite, chaque paire de nœuds est hachée récursivement jusqu'à ce que nous atteignions le noeud racine, qui est un hachage de tous les noeuds en dessous[84]. C'est ce principe qui est utilisé pour hacher les transactions dans le blockchaine Etherum. Un exemple de notre arbre de merkel est donné dans la figure 41.

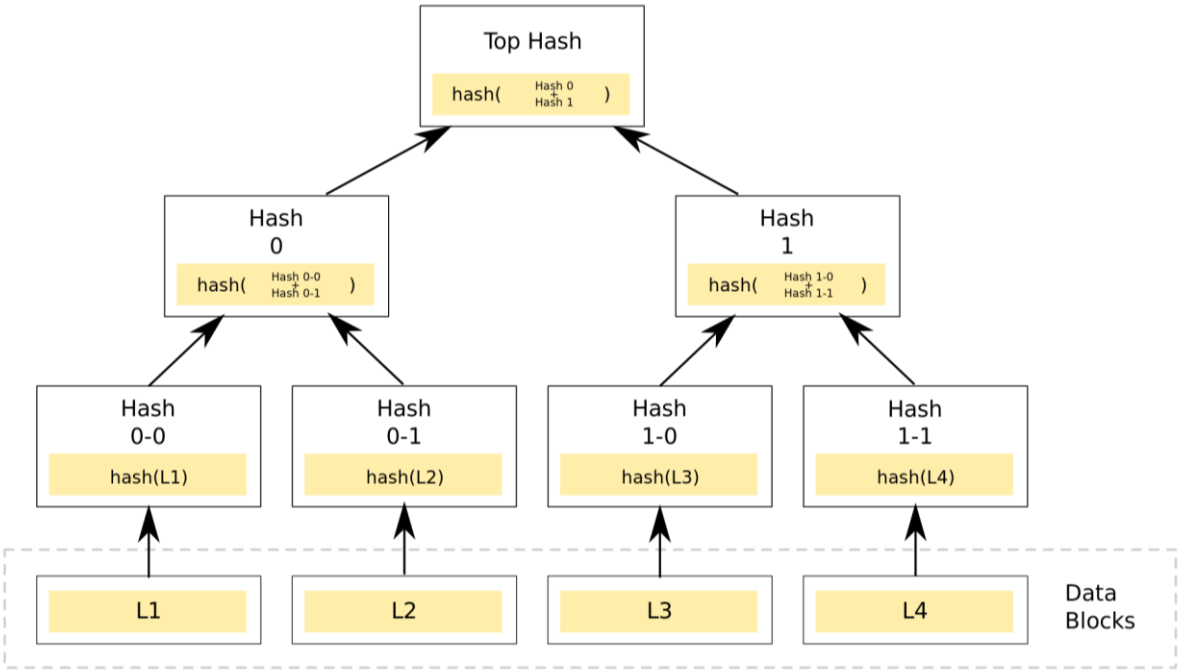

Figure 41: Merkel Tree

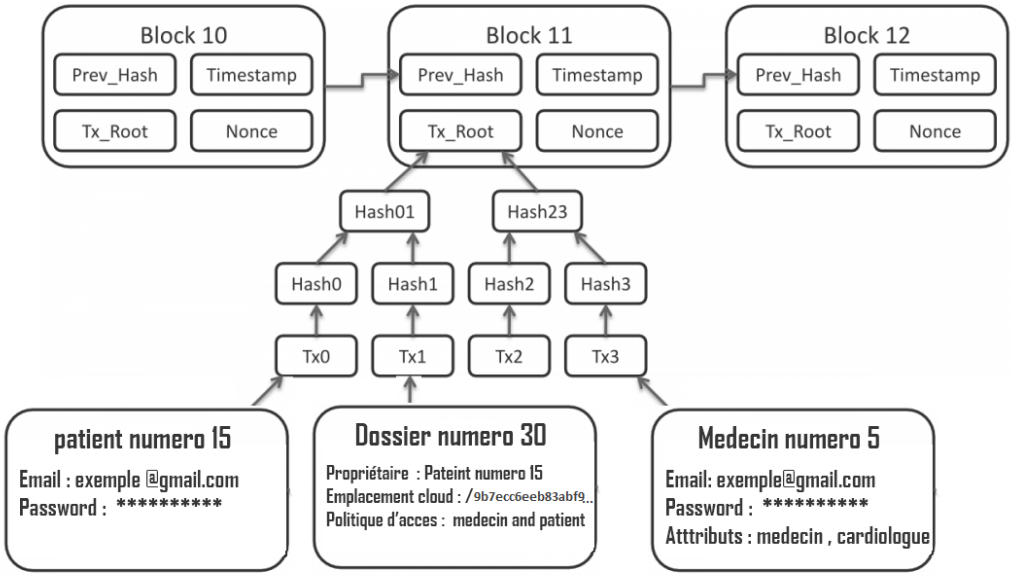

Figure 42: Exemple de notre arbre de merkel.

Dans cet exemple, Tx0 présente une transaction dans notre blockchain hash0 présente le hash de la transaction 0, hash 01 est le hash des hash0 et hash1 et ainsi de suite jusqu'au Tx_root.

On distingue trois transactions différentes :

- une transaction faite par le patient N° 15 où on trouve son email et mot de passe chiffrée.

- Une transaction faite sur le dossier médical N° 30 où on trouve son propriétaire, le hash du dossier chiffré et son emplacement dans le cloud.
- Une transaction faire par le médecin N°5 où on trouve son email et mot de passe chiffrée ainsi que ses attributs.

3 Implémentation du chiffrement CP-ABE :

Dans notre application, nous avons utilisé l'algorithme de CP-ABE crée par le Département d'informatique de l'université Johns Hopkins, Baltimore [85]. Ce dernier est un package Linux (cpabe-0.11.tar.gz, libbswabe-0.9.tar.gz[4]) qui doit être exécuté sur une machine(serveur) Linux à travers quatre commandes principales :

- cpabe-setup – génère une clé publique et une clé secrète principale.
- cpabe-keygen – génère une clé privée avec un ensemble donné d'attributs.
- cpabe-enc – crypte un fichier selon une politique, qui est une expression en termes d'attributs .
- cpabe-dec – déchiffre un fichier à l'aide d'une clé privée

Dans la figure 43. un exemple de l'utilisation du package Linux CP-ABE dans notre application

```
280  app.get('/GetDossier/:id', (req, res) => {
281    console.log("**** GET /GetDossier ****");
282    var x=req.params.id.split("_") ; var dossier_id=x[1]; var patinet_id=x[0];
283    truffle_connect.GetPatientDossier(patinet_id,dossier_id,(result) => {
284      console.log("TRANSACTION GetPatientDossier VALIDE ")
285      console.log(result)
286      const file = fs.createWriteStream("dossier.txt.cpabe");
287      https.get(ipfs_address+result[2], function(response) {
288        console.log("Recuperation du dossier avec le hash IPFS : "+result[2])
289        response.pipe(file);
290        exec("cpabe-dec pub_key a_private_key  dossier.txt.cpabe", (error, stdout, stderr) => {
291          if (error) {
292            console.log(`error: vous n'avez pas l'autorisation d'accéder a ce dossier ${error.message}`);
293            res.send("vous n'avez pas l'autorisation d'accéder a ce dossier" )
294            return;
295          }
296          if (stderr) {
297            console.log(`stderr: ${stderr}`);
298
299            res.send("error");
300            return;
301          }
302          console.log("FICHIER DECRYPTE , AVEC LA CLE SECRETE")
303          let rawdata = fs.readFileSync('dossier.txt');
304          var dossier = JSON.parse(rawdata);
305            res.render("FicheDeSuivi",{"dossier":dossier,"dossier_id":dossier_id,"patinet_id":patinet_id,"user_type":req.session.type})
306          });
307      });
308    });
309  });
```

Figure 43 : Utilisation CPABE dans l'application

D'abord, nous récupérons le hash du dossier (ligne 283) qui existe dans le blockchain ensuite nous téléchargeons le fichier existant à travers IPFS (ligne 287). Puis, nous exécutons cp-abe-dec (ligne 290) pour déchiffrer le fichier. Cela nécessite comme argument la clé publique et la clé secrète ainsi que le dossier chiffré. En cas erreur (la ligne 291), un message d'erreur sera affiché et le code sera interrompu (ligne 294) sinon la page contenant le dossier déchiffré sera affichée (la ligne 305).

4 Présentation de l'application

Dans cette partie, nous allons présenter les différentes interfaces de l'application

4.1 Page d'authentification

Dans cette page, l'utilisateur doit mettre les informations d'authentification requise et préciser s'il s'agit d'un patient ou médecin pour accéder à plateforme :

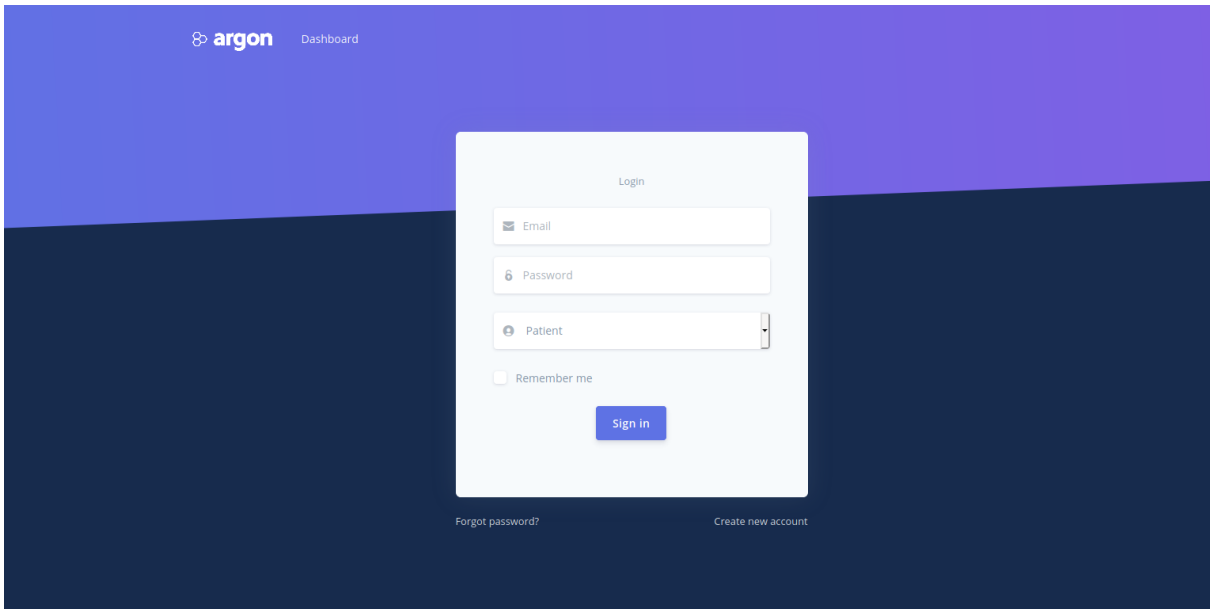

Figure 44: la page d'authentification

4.2 Profile

Dans cette partie, le patient peut saisir et modifier ses informations personnelles.

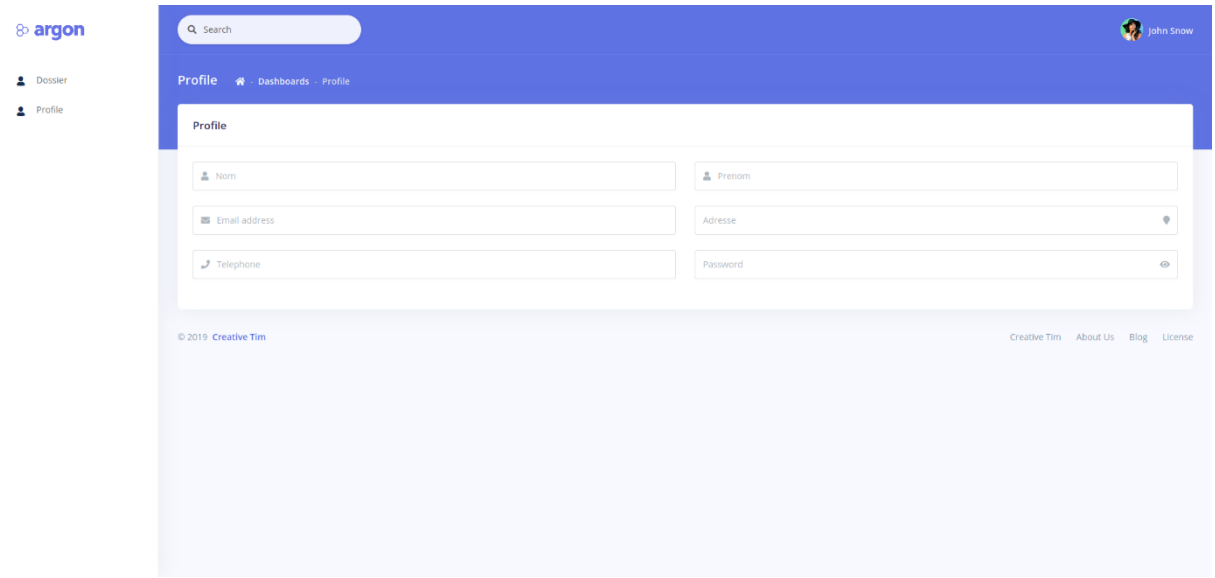

Figure 45: Page du Profile

4.3 Création des droits d'accès

Dans notre application, le patient peut créer les droits d'accès à son dossier médical

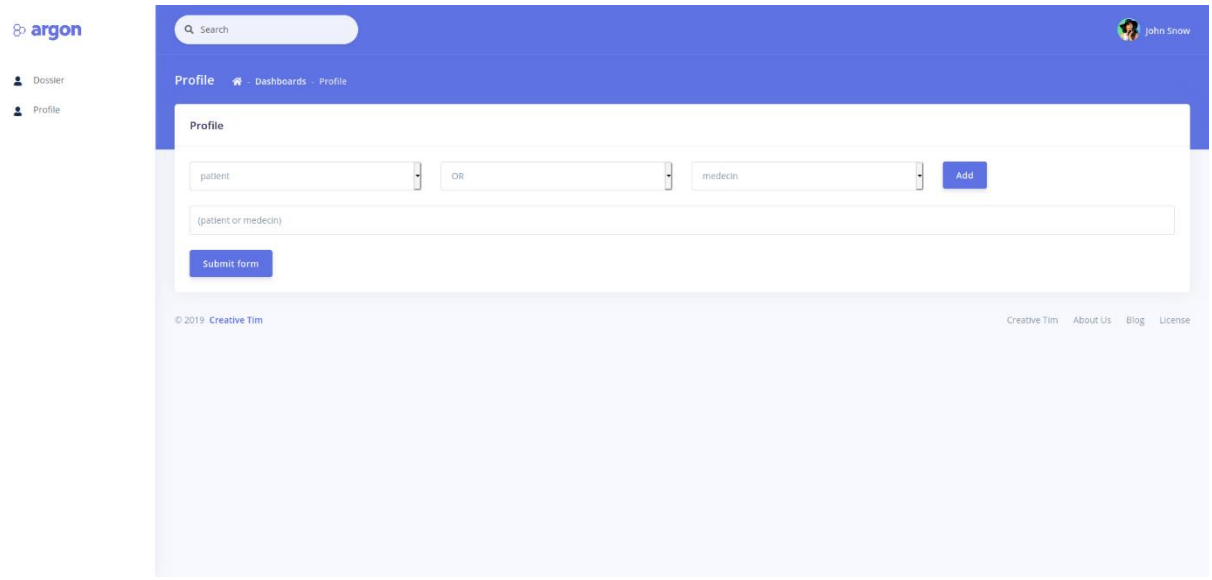

Figure 46: Création de droit d'accès

4.4 Ajouter une fiche de suivi

Le médecin peut ajouter des fiches de suivi au dossier du patient s'il a accès à ce dernier, La fiche de suivi peut inclure :

- Remarques
- Documents(analyse)
- Photos et vidéos (radio, scanner …etc.)

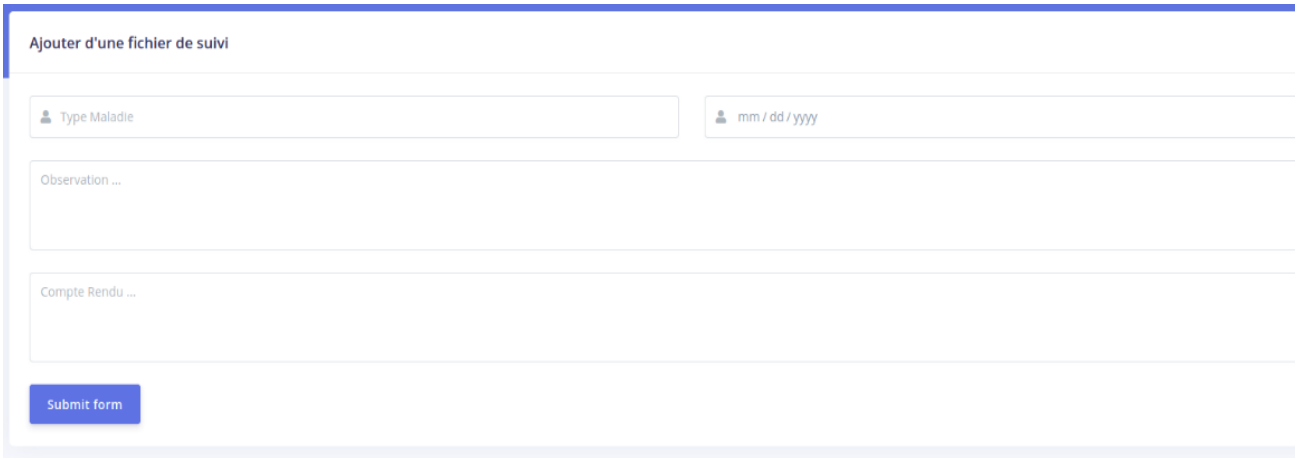

Figure 47: ajouter fiche de suivi

4.5. Consultation du Dossier médical

Dans cette page, nous présentons l'interface du dossier médical qui se compose de plusieurs fiches de suivi ordonné selon la date d'ajout.

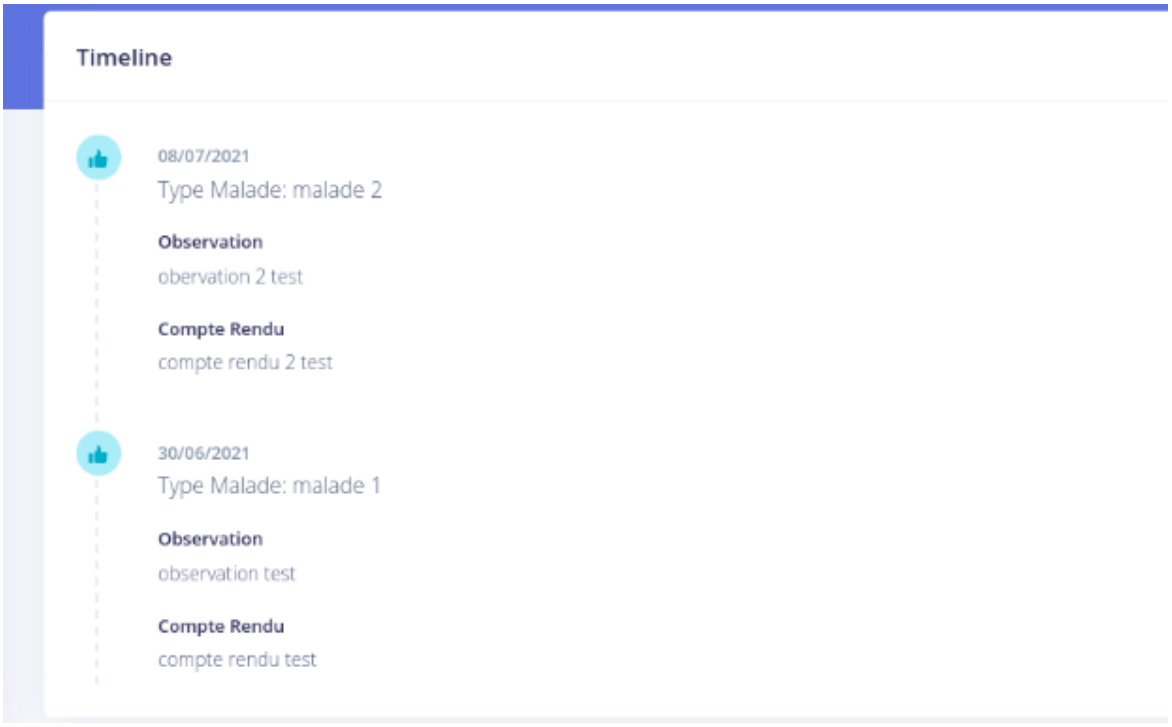

Figure 48: dossier médicale

4.6 Recherche patient

Dans cette interface, le médecin peut faire une recherche pour trouver un patient.

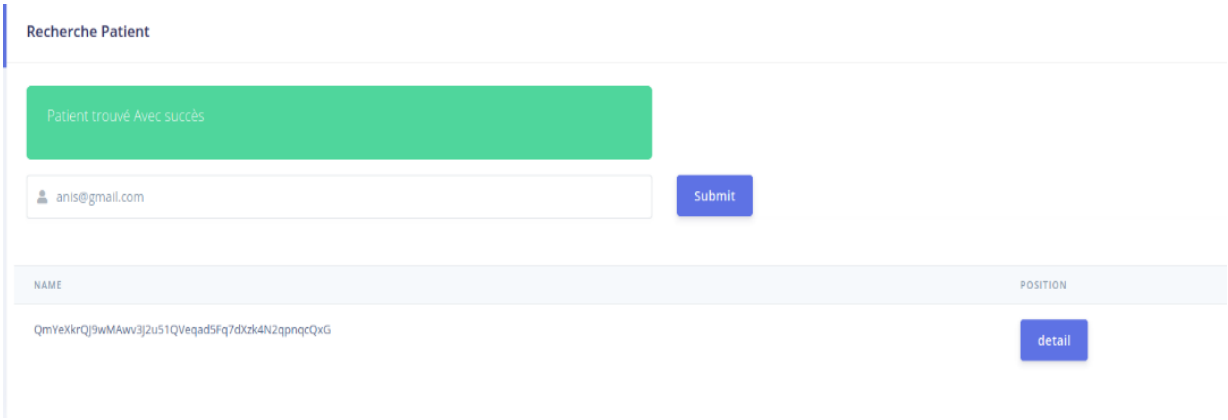

Figure 49: interface recherche du patient

4.7 Gestion des attributs par l'autorité de confiance

Dans cet espace, l'autorité de confiance définit les attributs du médecin pour un patient

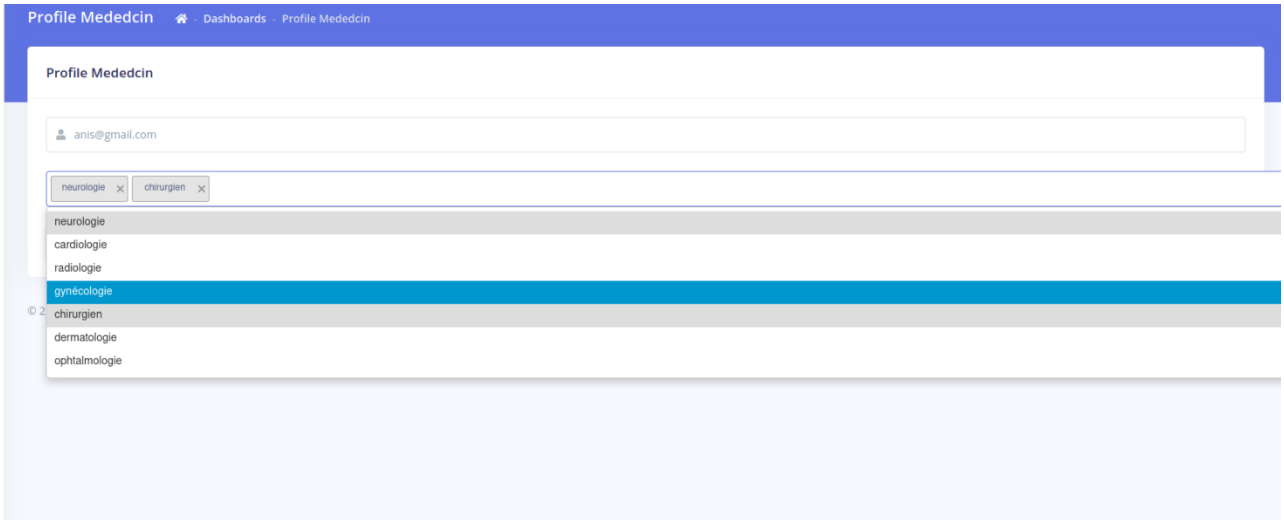

Figure 50: Gestion des Attributs par l'autorité de confiance

5 Conclusion

Dans ce chapitre, nous avons décrit les différentes technologies utilisées pour la réalisation de notre projet et leur utilisation en suivant notre conception. Nous avons utilisé le blockchain Ethereum pour stocker les informations d'authentification, les attributs utilisés dans le chiffrement CP-ABE et le hash des dossiers médicaux généré par le cloud IPFS. Autrement dit, nous avons achevé la création de la première version de notre application selon la conception proposée.

Conclusion Générale et Perspectives

L'objectif de ce travail est de trouver une solution à la problématique de la sécurité des données médicales stockées dans les applications e-santé afin de pouvoir les partager avec les utilisateurs autorisés tout en préservant la confidentialité, l'intégrité et leur vie privée.
Pour atteindre notre objectif, nous avons fait une étude bibliographique que nous avons répartie sur deux chapitres :

- Nous avons d'abord exploré les blockchains pour montrer leur utilisation toute en assurant la sécurité de nos informations. Puis, nous avons passé au cloud computing afin de remédier au problème de stockage des données médicales dans le domaine e-santé basé sur blockchain.
- Dans le chapitre 2, nous avons entamé le chiffrement et le contrôle d'accès pour rajouter une autre couche de sécurité aux données sauvegardées dans le cloud. Nous nous sommes intéressés au contrôle d'accès basé sur les attributs et le chiffrement CP-ABE.

Après, nous avons proposé une plateforme e-santé qui combine trois approches de sécurité :

- Un blockchain privée qui enregistre et valide toute les transactions de notre application
- Un cloud IPFS qui stocke nos dossiers de façon sécurisé
- Un contrôle d'accès ABAC avec l'algorithme de chiffrement par attribut CP-ABE qui chiffre les dossiers avant de les stocker dans le cloud.

En combinant ces approches, nous avons pu réaliser les objectifs de sécurité requis dans ce domaine :

- **La disponibilité :** avec la technologie blockchain et le cloud IPFS qui s'exécutent sur plusieurs nœuds nous assure une disponibilité très élevée de l'information comparée à d'autres technologies.
- **Contrôle d'accès :** permet de déterminer les utilisateurs autorisés à accéder à un dossier médical donné.
- **La confidentialité :** les données dans le cloud sont cryptées et seules les personnes autorisées peuvent y accéder.
- **L'intégrité :** de nos données est assurée grâce à la technologie blockchain et ses avantages dans l'immutabilité.
- **La non répudiation :** aucune transaction dans notre système ne peut être niée car le blockchain sauvegarde toute les transactions qui ne peuvent pas être altérées.
- **La traçabilité :** qui conserve les traces de l'état et des mouvements de l'information,

Pour mettre en place notre solution, nous avons implémenté nos approches en une application NodeJs en utilisant le blockchain Ethereum avec le cloud IPFS et une librairie pour le chiffrement CP-ABE. Notre application permet :

- à l'utilisateur de s'authentifier par nom d'utilisateur/email et mot de passe,

- à l'autorité de confiance de gérer les attributs des utilisateurs et les clés de chiffrement et de déchiffrement,
- au médecin d'ajouter des fiches de suivis. Ces dernières sont chiffrées avant d'être stockées dans le cloud.
- au patient de consulter son dossier médical et définir les droits d'accès à ce dossier.

Toutefois, plusieurs améliorations qui peuvent y être apportées à notre plateforme, à titre d'exemple :

- Ajouter d'autres acteurs à notre application telle que les infirmiers, aide-soignant, responsable légal du patient et bien d'autre,
- Ajouter la possibilité d'importation des documents tels que les analyses, radios, IRM…,
- Ajouter à notre réseau blockchain différents hôpitaux de différent pays, sans crainte sur la confidentialité de nos données.

Bibliographie :

[1] Jigna J. Hathaliya, Sudeep Tanwar, An exhaustive survey on security and privacy issues in Healthcare 4.0, Computer Communications, Volume 153,2020,Pages 311-335,ISSN 0140-3664.

[2] K. Hayrinen, K. Saranto, P. Nykanen, Definition, structure, content, use and impacts of electronic health records: a review of the research literature Int J Med Inform, 77 (5) (2008), pp. 291-304

[3] Sánchez JL, Savin S, Vasileva V. Key success factors in implementing electronic medical records in University Hospital of Rennes: Rennes: ENSP; 2005. p. 1–59

[4] F. Lau, M. Price, J. Boyd, C. Partridge, H. Bell, R. Raworth Impact of electronic medical record on physician practice in office settings: a systematic review BMC Med Inform Decis Mak, 12 (1) (2012), pp. 1-10

[5] R.H. Miller, I. Sim Physicians' use of electronic medical records: barriers and solutions Health Aff, 23 (2) (2004), pp. 116-126

[6] A. Boonstra, M. Broekhuis Barriers to the acceptance of electronic medical records by physicians from systematic review to taxonomy and interventions BMC health Serv Res, 10 (1) (2010), pp. 231-247

[7] D. Blumenthal, M. Tavenner The "meaningful use" regulation for electronic health records New Engl J Med, 363 (6) (2010), pp. 501-504

[8] Hsiao CJ, Hing E. Use and characteristics of electronic health record systems among office-based physician practices: United States, 2001–2013. In: Services USDoHaH, editor. Hyattsville, Maryland: National Center for Health Statistics; 2014.

[9] C. Lin, E.W. Karlson, D. Dligach, M.P. Ramirez, T.A. Miller, H. Mo, et al. Automatic identification of methotrexate-induced liver toxicity in patients with rheumatoid arthritis from the electronic medical record J Am Med Inform Assoc, 22 (e1) (2014), pp. 151-161

[10] K.M. Krysko, N.M. Ivers, J. Young, P. O'Connor, K. Tu Identifying individuals with multiple sclerosis in an electronic medical record Mult Scler J, 21 (2) (2015), pp. 217-224

[11] Tsipi Heart, Ofir Ben-Assuli, Itamar Shabtai, A review of PHR, EMR and EHR integration: A more personalized healthcare and public health policy, Health Policy and Technology, Volume 6, Issue 1, 2017, Pages 20-25, ISSN 2211-8837, https://doi.org/10.1016/j.hlpt.2016.08.002.

[12] E.A. Coleman, J.D. Smith, J.C. Frank, S.J. Min, C. Parry, A.M. Kramer Preparing patients and caregivers to participate in care delivered across settings: the care transitions intervention J Am Geriatr Soc, 52 (11) (2004), pp. 1817-1825

[13] P.C. Tang, J.S. Ash, D.W. Bates, J.M. Overhage, D.Z. Sands Personal health records: definitions, benefits, and strategies for overcoming barriers to adoption J Am Med Inform Assoc, 13 (2) (2006), pp. 121-126

[14] T. McKeown The role of medicine. Dream, mirage or nemesis? Basil Blackwell Publisher Ltd., Oxford, England (1979)

[15] P.E. Kummervold, C.E. Chronaki, B. Lausen, H.U. Prokosch, J. Rasmussen, S. Santana, et al. eHealth trends in Europe 2005-2007: a population-based survey J Med Internet Res, 10 (2008), p. 4

[16] N. Archer, U. Fevrier-Thomas, C. Lokker, K.A. McKibbon, S.E. Straus Personal health records: a scoping review J Am Med Inform Assoc, 18 (4) (2011), pp. 515-522

[17]C.L. Goldzweig, G. Orshansky, N.M. Paige, A.A. Towfigh, D.A. Haggstrom, I. Miake-Lye, et al. Electronic patient portals: evidence on health outcomes, satisfaction, efficiency, and attitudesa systematic review Ann Intern Med, 159 (10) (2013), pp. 677-687

[18] S.J. Czaja, C. Zarcadoolas, W.L. Vaughon, C.C. Lee, M.L. Rockoff, J. Levy The usability of electronic personal health record systems for an underserved adult population Hum Factors: J Hum Factors Ergon Soc, 57 (3) (2014), pp. 491-506

[19] M. Kaur, M. Murtaza and M. Habbal, "Post study of Blockchain in smart health environment," 2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA), 2020, pp. 1-4, doi: 10.1109/CITISIA50690.2020.9371819.

[20] Hadrien Boé, LES BASES DE LA BLOCKCHAIN, https://blog.eleven-labs.com/fr/bases-blockchain/ consulter le 06/07/2021

[21] A. Shahnaz, U. Qamar and A. Khalid, "Using Blockchain for Electronic Health Records," in IEEE Access, vol. 7, pp. 147782-147795, 2019, doi:10.1109/ACCESS.2019.2946373.

[22] Gilbert S., Lynch N. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services ACM SIGACT News, 33 (2) (2002), pp. 51-59

[23] Comprendre la blockchain, Livre blanc sous licence Creative Commons, édité par uchange.co, janvier 2016.

[24] Delahaye, Jean-Paul. "Les blockchains, clefs d'un nouveau monde.", pour la Science 449 (2015) :80-85.

[25] Claire Fénéron Plisson, "La blockchain, un bouleversement économique, juridique voire sociétal", I2D ? Information, données & documents,(Volume 54), p. 20-22, mars 2017.

[26] Thomas Dupont, a Blockchain : introduction et applications à, Etopia, 09/04/2018

[27] C. Hammerschmidt, Consensus in Blockchain systems. In Short, Available: https://medium.com/@chrshmmmr/consensus-in-blockchain-systems-in-short-691fc7d1fefe, Accessed on 5th of February 2018.

[28] M. Correia, G.S. Veronese, N.F. Neves, P. Verissimo Byzantine consensus in asynchronous message-passing systems: a survey Int. J. Crit. Comput. Based Syst., 2 (2) (2011), pp. 141-161

[29] K. Yeow, A. Gani, R.W. Ahmad, J.J.P.C. Rodrigues, K. Ko Decentralized consensus for edge-centric internet of things: a review, taxonomy, and research issues IEEE Access, vol. 6 (2018), pp. 1513-1524

[30]https://coinrevolution.com/fr/what-is-the-difference-between-a-blockchain-and-a-database/ consulter le 04/05/2021

[31] MICHAEL J. FISCHER, NANCY A. LYNCH, MICHAEL S. PATERSON Impossibility of Distributed Consensus with One Faulty Process, Journal of the Association for Computing Machinery, Vol. 32, No. 2, April 1985

[32]F. Hawlitschek, B. Notheisen, T. Teubner, The limits of trust-free systems: a literature review on blockchain technology and trust in the sharing economy Electron. Commer. Res. Appl., 29 (2018).

[33]K. Heires The risks and rewards of blockchain technology Risk Management, 63 (2) (2016), pp. 4-7 http://www.rmmagazine.com/2016/03/01/the-risks-and-rewards-of-blockchain-technology/

[34] E. Mik Smart contracts: terminology, technical limitations and real world complexity Law Innov. Technol., 9 (2) (2017), pp. 269-300, 10.1080/17579961.2017.1378468

[35] P. Boucher, How blockchain technology could change our lives: in-depth analysis, European Parliament, 2017,

[36] E.B. Hamida, K.L. Brousmiche, H. Levard, E. Thea, Blockchain for enterprise: Overview, opportunities and challenges, in: The Thirteenth International Conference on Wireless and Mobile Communications (ICWMC 2017), 2017, https://hal.archives-ouvertes.fr/hal-01591859/document,

[37] J. Ronald and J. Krotoszynski, The polysemy of privacy, Indiana Law J., 88 (2013) pp. 881–918, http://ilj.law.indiana.edu/articles/8-Krotoszynski.pdf,

[38] G. Neyer and B. Geva, Blockchain and payment systems: what are the benefits and costs?, Journal of Payments Strategy & Systems, (2017)

[39] Liu, W., Zhu, S.S., Mundie, T. and Krieger, U., 2017, October. Advanced block-chain architecture for e-health systems. In e-Health Networking, Applications and Services (Healthcom), 2017 IEEE 19th International Conference on (pp. 1-6). IEEE.

[40]M. Engelhardt, "Hitching Healthcare to the Chain: An Introduction to Blockchain Technology in the Healthcare Sector", Technology Innovation Management Review, vol. 7, no. 10, pp. 22-34, 2017.

[41] A. Siyal, A. Junejo, M. Zawish, K. Ahmed, A. Khalil and G. Soursou, "Applications of Blockchain Technology in Medicine and Healthcare: Challenges and Future Perspectives", Cryptography, vol. 3, no. 1, pp. 2-3, 2019.

[42] Y. Chen, S. Ding, Z. Xu, H. Zheng and S. Yang, "Blockchain-Based Medical Records Secure Storage and Medical Service Framework", Journal of Medical Systems, vol. 43, no. 1, 2018.

[43] X. Yue, H. Wang, D. Jin, M. Li and W. Jiang, "Healthcare Data Gateways: Found Healthcare Intelligence on Blockchain with Novel Privacy Risk Control", Journal of Medical Systems, vol. 40, no. 10, 2016.

[44] S. Theodouli, K. Arakliotis, K. Moschou, D. Votis and Tzo-varas, "On the design of a Blockchain-based system to facilitate Healthcare Data Sharing", 17th IEEE International Conference On Trust Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pp. 1374-1379, 2018.

[45] A. Fusco, G. Dicuonzo, V. Dell'Atti and M. Tatullo, "Blockchain in Healthcare: Insights on COVID-19", International Journal of Environmental Research and Public Health, vol. 17, no. 19, pp. 7167, 2020.

[46] Sivagami, S., Revathy, D. and Nithyabharathi, L., 2016. Smart Health Care System Implemented Using IoT. International Journal of Contemporary Research in Computer Science and Technology, 2(3).

[47] Theodouli, A., Arakliotis, S., Moschou, K., Votis, K. and Tzovaras, D., 2018, August. On the design of a Blockchain-based system to facilitate Healthcare Data Sharing. In 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE) (pp. 1374-1379). IEEE.

[48] Wang, S., Wang, J., Wang, X., Qiu, T., Yuan, Y., Ouyang, L., Guo, Y. and Wang, F.Y., 2018. Blockchain-Powered Parallel Healthcare Systems Based on the ACP Approach. IEEE Transactions on Computational Social Systems, (99), pp.1-9.

[49] Alhadhrami, Z., Alghfeli, S., Alghfeli, M., Abedlla, J.A. and Shuaib, K., 2017, November. Introducing blockchains for healthcare. In Electrical and Computing Technologies and Applications (ICECTA), 2017 International Conference on (pp. 1-4). IEEE

[50] Liang, X., Zhao, J., Shetty, S., Liu, J. and Li, D., 2017, October. Integrating blockchain for data sharing and collaboration in mobile healthcare applications. In Personal, Indoor, and Mobile Radio

[51] P. Mell et al., "Google Application Engine Introduction," Futur. Gener. Comput. Syst., vol. 25, no. 6, p. 17, 2011.

[52] 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

[53] Coles M., Landrum R. (2009) Hashing. In: Expert SQL Server 2008 Encryption. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4302-3365-7_7

[54] Craig Wright, in The IT Regulatory and Standards Compliance Handbook, 2008

[55] Chirag Langaliya , Rajanikanth Aluvalu , Enhancing Cloud Security through Access Control Models: A Survey ,International Journal of Computer Applications (0975 – 8887) Volume 112 – No. 7, February 2015

[56] Samarati P., de Vimercati S.C. (2001) Access Control: Policies, Models, and Mechanisms. In: Focardi R., Gorrieri R. (eds) Foundations of Security Analysis and Design. FOSAD 2000. Lecture Notes in Computer Science, vol 2171. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45608-2_3

[57] Ferraiolo, Sandhu, Gravila, Kuhn and Chandramouli, "Proposed NIST Standard for Role-Based Access Control," ACM Transactions on Information and System Security, vol. 4, no. 3, pp. 222- 274, 2001.

[58] Alban Gabillon. Contrôler les accès aux données numériques. La Revue de l'Electricité et de l'Electronique, Société de l'Électricité, de l'Électronique et des Technologies de l'Information et de la Communication, 2013, 12 p. ffhal-02108021f

[59] RBAC vs. ABAC: What's the Difference? https://www.dnsstuff.com/rbac-vs-abac-access-control consulter le 15/06/2021

[60] Wattana Viriyasitavata,Danupol Hoonsoponb, Blockchain characteristics and consensus in modern business processes, Journal of Industrial Information Integration, Volume 13, 2019, Pages 32-39, ISSN 2452-414X, https://doi.org/10.1016/j.jii.2018.07.004.

[61] Applications e-santé, le contrôle des données personnelles un enjeu majeur pour la protection de la vie privée, Youcef OULD YAHIA1,2 et Pierre PARADINAS1. 1Conservatoire National des Arts et Métiers, Paris, France 2 École Polytechnique de Montréal, Canada.

[62] International Journal of Network Security, Vol.15, No.4, PP.231-240, July 2013 "A Survey on Attribute-based Encryption Schemes of Access Control in Cloud Environments". Cheng-Chi Lee1, Pei-Shan Chung2, and Min-Shiang Hwang3.

[63] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext Policy Attribute-based Encryption", in Proceedings of IEEE Symposium on Security and Privacy, (2007).

[64] Sahai and B. Waters, "Fuzzy Identity-based Encryption", Advances in Cryptography" V EUROCRYPT, vol. 3494, pp. 457-473.

[65] International Journal of Cloud-Computing and Super-Computing Vol.2, No.2 (2015), pp.1-6 http://dx.doi.org/10.21742/ijcs.2015.2.2.01: Attribute-based Encryption for Electronic Health Records in a Cloud Computing Environment, Emmanuel KusiAchampong1,2and Clement Dzidonu. 1 Department of Medical Education and IT, University of Cape Coast, Cape Coast . 2 Accra Institute of Technology.

[66] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel and W. Jonker, "Mediated Ciphertext Policy Attribute-based Encryption and its Application", Information Security Applications, vol. 5932, (2009), pp. 309-323.

[67] L. Ibraimi, Q. Tang, P. Hartel and W. Jonker, "Efficient and Provable Secure Ciphertext Policy Attribute-based Encryption Schemes", in Proceedings of the Information Security Practice and Experience, (2009).

[68] 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.

[69]2012 Fourth International Conference on Intelligent Networking and Collaborative,Systems, « Implementing a Personal Health Record Cloud Platform Using Ciphertext-Policy Attribute-Based Encryption". Changji Wang, Xuan Liu, Wentao Li, School of Information Science and Technology, Guangdong Province Information Security Key Laboratory, Sun Yat-sen University, Guangzhou 510275, China.

[70] “Designing a Secure Cloud-Based EHR System using Ciphertext-Policy Attribute-Based ncryption » SuhairAlshehri, StanisławRadziszowski, and Rajendra K. Raj. Golisano College of Computing & Information Sciences, Rochester Institute of Technology, Rochester, New York 14623, USA Conference Paper: April 2012.

[71] B. Raja Sekhar,B. Sunil Kumar, L. Swathi Reddy, V. PoornaChandar. “CP-ABE Based Encryption for Secured Cloud Storage Access”.International Journal of Scientific & Engineering Research, Volume 3, Issue 9, September-2012 ISSN 2229-5518

[72] Qi Yuan, Chunguang Ma, Junyu Lin« Fine-Grained Access Control for Big Data Based on CP-ABE in Cloud Computing».InternationalConference of Young Computer Scientists, Engineers and Educators, ICYCSEE 2015 Intelligent Computation in Big Data Era pp344-352.

[73] Kan Yang, XiaohuaJia, Kui Ren« Attribute-based Fine-Grained access Control with Efficient Revocation in Cloud Storage Systems».

[74] Benet, J. (2014). Ipfs-content addressed, versioned, p2p €le system. arXiv preprint arXiv:1407.3561.

[75] Politou, E.; Alepis, E.; Patsakis, C. Delegated content erasure in IPFS.Future Gener. Comput. Syst.2020,112, 956–964

[76] Chaitali Acharya, How can Blockchain help in Background Check?, https://harbinger-systems.com/blog/2018/11/how-can-blockchain-help-in-background-check/

[77] Web3.Js , https://web3js.readthedocs.io/en/v1.3.4/

[78] introduction to the SHA-256 hash function https://steemit.com/cryptocurrency/@f4tca7/introduction-to-the-sha-256-hash-function

[79] Design of SHA-3 (Keccak)

[80] Ethereum Documentation, https://ethereum.org/en/developers/docs/

[81]NodeJs , https://nodejs.org/en/

[82] Gregory. How to build Blockchain App-Ethereum ,

https://www.dappuniversity.com/articles/blockchain-app-tutorial, consulter le 20/06/2021

[83]Truffle.Js , https://www.trufflesuite.com/

[84] Merkle Tree. Brilliant.org. Retrieved 20:37, June 21, 2021, from https://brilliant.org/wiki/merkle-tree/

[85] John Bethencourt , Amit Sahai, Brent Waters, Ciphertext-Policy Attribute-Based Encryption, http://acsc.cs.utexas.edu/cpabe/ consulted on 04/04/2021