Ministère de l'enseignement supérieur et de la recherche

scientifiqueUniversité Saad Dahleb Blida-1-

Faculté des sciences

Département d'informatique

Projet de fin d'études en sécurité des systèmes d'information en vue del'obtention du diplôme de Master

Mémoire réalisé par : REBAHI Nour Elhouda

TEHAR Asma

Réalisation d'une plateforme de gestion des dossiers médicaux en utilisant la blockchain et le contrôle d'accès basé sur les attributs ABAC

Organisme d'accueil centre de recherches sur l'information scientifique et technique(CERIST)

Présenté le10/07/2023 Devant le jury composé de :

Mme N.Boustia promotrice

Mr S.Hadjar encadreur

Mme S.Oukid présidente

Mme S.Aroussi examinatrice

Année universitaire 2022-2023

Remerciements

Au moment de conclure cette étape importante de notre parcours académique, nous tenons à exprimer notre profonde gratitude envers Allah le Tout-Puissant, source de toute guidance et de tout succès, pour nous avoir accordé cette opportunité d'accomplir ce mémoire. Sa bénédiction et Sa protection ont été des soutiens constants tout au long de notre parcours.

Nous souhaitons également exprimer notre gratitude envers nos familles et nos proches qui ont été à nos côtés, nous soutenant de manière inconditionnelle. Votre amour, votre soutien et vos encouragements ont été d'une importance capitale dans notre réussite. Votre présence et votre soutienconstant nous ont donné la force et la motivation nécessaires pour surmonterles défis rencontrés.

Un remerciement tout particulier est adressé à notre encadreur, Mr S. HADJAR, pour ses conseils éclairés, son expertise et son soutien tout au long de la réalisation de ce mémoire. Votre accompagnement attentif et vos précieux conseils ont joué un rôle essentiel dans notre réussite.

Nous exprimons également notre reconnaissance envers nos enseignants et l'ensemble du corps professoral qui nous ont transmis leurs connaissances et leur expertise. Votre dévouement à l'enseignement et votre volonté de partager votre savoir ont grandement contribué à notre formation.

Enfin, nous remercions tous ceux qui, de près ou de loin, ont contribué à notre parcours académique et à la réalisation de ce mémoire. Votre soutien, vos conseils et votre contribution ont été d'une valeur inestimable.

Nous ne saurions exprimer pleinement la profondeur de notre gratitude envers chacun d'entre vous. Vos encouragements et votre présence bienveillante ont été une source de motivation et de réconfort tout au long denotre cheminement.

Nous vous remercions du fond du cœur pour votre soutien et votre confiance.

Résumé

La sécurité et la confidentialité des données médicales représentent un défi majeur. Les informations sensibles telles que les diagnostics, les ordonnances et les données personnelles des patients sont généralement stockées de manière centralisée, ce qui compromet leur niveau deconfidentialité. Dans notre proposition, nous avons développé une plateforme de gestion des données médicales basée sur la technologie blockchain pour garantir la confidentialité, l'intégrité et la disponibilité de ces données. Pour résoudre les problèmes de stockage liés au volume important des données médicales, nous avons configuré des nœuds complets avec une capacité de stockage significative. Cela renforce la sécurité des données et permet d'éviter les problèmes de stockage centralisé. De plus, nous avons intégré un modèle de contrôle d'accès basé sur les attributs ABAC et une méthode de chiffrement AES afin de chiffrer les données médicales et de préserver la vie privée des patients. Ce modèle facilite également l'accès aux données autorisées. En combinant ces technologies, notre plateforme assure la cohérence, l'intégrité, la disponibilité et la confidentialité des données médicales.Nous avons également réalisé une étude théorique sur les hôpitaux intelligents et les dossiers médicaux électroniques MDE en vue d'une éventuelle application de notre solution dans la gestion des données médicales. Cette démarche vise à étendre les avantages de notre plateforme à un environnement hospitalier connecté et intelligent.

Mots clés: Blockchain, ABAC, AES, MDE, smart hospitals.

Abstract

The security and confidentiality of medical data is a major challenge. Sensitive information such as diagnoses, prescriptions and patients' personal data are usually stored centrally, which compromises their level of confidentiality. In our proposal, we have developed a medical data management platform based on blockchain technology to guarantee the confidentiality, integrity andavailability of this data. To solve the storage problems associated with the large volume of medical data, we have configured complete nodes with significant storage capacity. This reinforces data security and avoids the problems of centralised storage. In addition, we have integrated an access control model based on attributs ABAC and encryption method AES to encrypt medical data and preserve patient privacy. This model also facilitates access to authorised data. By combining these technologies, our platform guarantees the consistency, integrity, availability and confidentiality of medical data. We have also carried out a theoretical study on intelligent hospitals and electronic medical records with aview to the possible application of our solution to the management of medical data. This approach aims to extend the benefits of our platform to a connected and intelligent hospital environment.

Keywords: Blockchain, nodes, ABAC, AES, EDM, smart hospitals.

ملخص

يمثل أمن وسرية البيانات الطبية تحديًا كبيرًا. عادةً ما يتم تخزين المعلومات الحساسة مثل التشخيصات والوصفات الطبية وبيانات المرضى الشخصية مركزيًا، مما يعرض مستوى سريتها للخطر. في اقتراحنا، قمنا بتطوير منصة لإدارة البيانات الطبية لضمان سرية هذه البيانات وسلامتها وتوافرها. لحل مشكلات التخزين المتعلقة بالحجم blockchain تعتمد على تقنية الكبير للبيانات الطبية، قمنا بتكوين العقد الكاملة ذات سعة تخزين كبيرة. وهذا يعزز أمن البيانات ويتجنب مشاكل التخزين لتشفير AES وطريقة تشفير ABAC المركزية. بالإضافة إلى ذلك، قمنا بدمج نموذج التحكم في الوصول القائم على سمات البيانات الطبية والحفاظ على خصوصية المريض. يسهل هذا النموذج أيضًا الوصول إلى البيانات المعتمدة. ومن خلال الجمع بين هذه التقنيات، تضمن منصتنا اتساق البيانات الطبية وسلامتها وتوافرها وسريتها. أجرينا أيضًا دراسة نظرية على من أجل إمكانية تطبيق حلنا في إدارة البيانات الطبية. ويهدف MDE المستشفيات الذكية والسجلات الطبية الإلكترونية هذا النهج إلى توسيع فوائد منصتنا لتشمل بيئة مستشفى متصلة وذكية.

الكلمات المفتاحية: Blockchain ،ABAC ،AES ،MDEالمستشفيات الذكية ،.

Table des matières

Liste des figures

Liste des tableaux

Liste des acronymes

AES	Advanced Encryption Standard
AC	Autorité de Confiance
ABE	Attribute-Based Encryption
ABAC	Attribute Based Access Control
BFT	Byzantine Fault Tolerance
CP-ABE	Ciphertext Policy Attribute-Based Encryption
DAPP	Decentralized Application
DME	Dossier Medical Electronic
EVM	Ethereum Virtual Machine ETH Ethereum
IPFS	Interplanetary File System
NIST	National Institute of Standards and Technology
POW	Proof of work
POS	Proof of Stack
POA	Proof of authority
RFID	Radio-frequency identification
XOR	Exclusive OR

Introduction générale

La gestion des données médicales au sein des hôpitaux et des centres de santé constitue un défi majeur dans le domaine de la santé. Souvent, ces données sont hébergées sur des serveurs centraux, avec des applications frontales gérées par des administrateurs, entraînant des problèmes tels que la consultation inefficace des informations médicales, la nécessité pour les patients de transporter des documents papier entre les services, et des risques de perte, d'altération ou de vol de données.

Face à ces lacunes de gestion des données médicales, une solution prometteuse émerge : la blockchain. Cette technologie offre la possibilité de sécuriser les informations des patients, de garantir leur confidentialité, et de favoriser l'interopérabilité des données de santé. La nature décentralisée et immuable de la blockchain permet d'assurer l'intégrité des données médicales à travers l'ensemble des systèmes d'information, tout en limitant l'accès aux données uniquement aux personnes autorisées.

Ce mémoire se propose d'explorer en détail l'application de la blockchain dans le domaine de la gestion des données médicales. Il s'articule autour de trois chapitres clés : une étude théorique sur les concepts fondamentaux de la blockchain et son application aux dossiers médicaux électroniques, la conception détaillée d'un système basé sur cette technologie, et enfin, la mise en œuvre concrète du système avec des détails sur le développement, la configuration, et les tests réalisés. En conclusion, nous récapitulerons les résultats obtenus, mettrons en évidence les avantages et les limitations de cette solution, tout en proposant des perspectives d'avenir pour l'utilisation de la blockchain dans le secteur de la santé.

Comment améliorer la gestion des données médicales dans les hôpitaux et les centres de soins, afin de résoudre les problèmes liés à la gestion actuelle des données médicales, tels que la perte de données, l'altération, le vol et les droits d'accès incorrects, et offrir une solution plus efficace et rentable pour stocker, gérer et échanger les données médicales entre les différents acteurs de la santé ?

Chapitre I : Etat de l'art

1. INTRODUCTION

Ce premier chapitre aborde deux sujets majeurs dans le domaine des technologies émergentes. Tout d'abord, nous nous penchons sur la blockchain, une technologie qui englobe à la fois une promesse sociale et une nouvelle approche technologique. Initialement développée pour le système d'enregistrement des crypto-monnaies comme Bitcoin, la blockchain est désormais utilisée dans diverses applications, offrant une base de données distribuée pour partager et enregistrer des transactions.

Ensuite, nous explorons le rôle crucial du contrôle d'accès dans la sécurité des données. Le contrôle d'accès détermine qui est autorisé à accéder et à utiliser les informations et les ressources d'une entreprise, garantissant ainsi que seuls les utilisateurs légitimes disposent d'un accès approprié. De plus, le contrôle d'accès peut être appliqué pour restreindre l'accès physique à des installations spécifiques.

Passant à un autre domaine, nous examinons l'impact des innovations technologiques numériques dans le secteur de la santé. La transformation numérique rapide dans ce domaine vise à optimiser les systèmes de gestion des soins de santé a donné lieu à l'émergence des hôpitaux intelligents et des dossiers médicaux électroniques. Ces innovations optimisent les systèmes de gestion des soins de santé en facilitant l'accès aux informations médicales des patients de manière sécurisée et en favorisant une meilleure coordination des soins.

Ce chapitre fournit donc une vision globale de la technologie blockchain, du contrôle d'accès et des dossiers médicaux électroniques, en soulignant leur importance et leurs applications dans différents secteurs, et surtout la santé.

2. Blockchain

2.1. Définition

La notion de Blockchain est apparue en 2008 lors de la création du bitcoin, comme son cas d'usage le plus connu. On trouve plusieurs propositions pour la définir, parmi ces définitions on cite les suivantes :

- Une Blockchain est une base de données transactionnelle distribuée,

comparable à un grand livre comptable décentralisé et partagé, qui stocke et transfère de la valeur ou des données via Internet, de façon transparente, sécurisée, et autonome sans organe central de contrôle. [1]

Propriétés	Blockchain	Base de données traditionnelle
Opérations	Seulement des opérations d'insertion	Peut effectuer des opérations CRUD
Réplication	Réplication complète du bloc sur chaque pair	Maître esclave multi-maître
Consensus	La majorité des pairs s'accordent sur le résultat des transactions	Transactions distribuées (validation en 2 phases)

Tableau 1 Comparaison entre Base de données et Blockchain

2.2. Domaines d'application de la blockchain

La blockchain trouve de plus en plus d'applications dans divers domaines, notamment la finance, la chaîne d'approvisionnement, le vote électronique et la santé. Dans le domaine financier, elle est utilisée pour faciliter les transactions et réduire les coûts de gestion des données. Elle permet la création de systèmes de paiement électronique décentralisés et sécurisés [2], ainsi que le développement de contrats intelligents et de plateformes de prêt décentralisées. Dans la chaîne d'approvisionnement, la blockchain offre une traçabilité complète et une transparence accrue en permettant de suivre les produits à chaque étape. Des entreprises comme Walmart et Maersk utilisent la blockchain pour suivre les produits frais et les conteneurs, respectivement. Cette technologie améliore la transparence et l'efficacité tout en réduisant les coûts [3]. Dans le domaine du vote électronique, la blockchain assure l'intégrité, l'authenticité et l'anonymat des votes enregistrés [4]. Enfin, dans le secteur de la santé, la blockchain permet la gestion sécurisée et décentralisée des données de santé, la collaboration entre les acteurs de la chaîne de soins et l'émergence de nouveaux modèles économiques. Cependant, des défis réglementaires et de protection de la vie privée doivent être relevés. Des recherches supplémentaires sont nécessaires pour évaluer pleinement le potentiel de la blockchain dans le domaine de la santé [5].

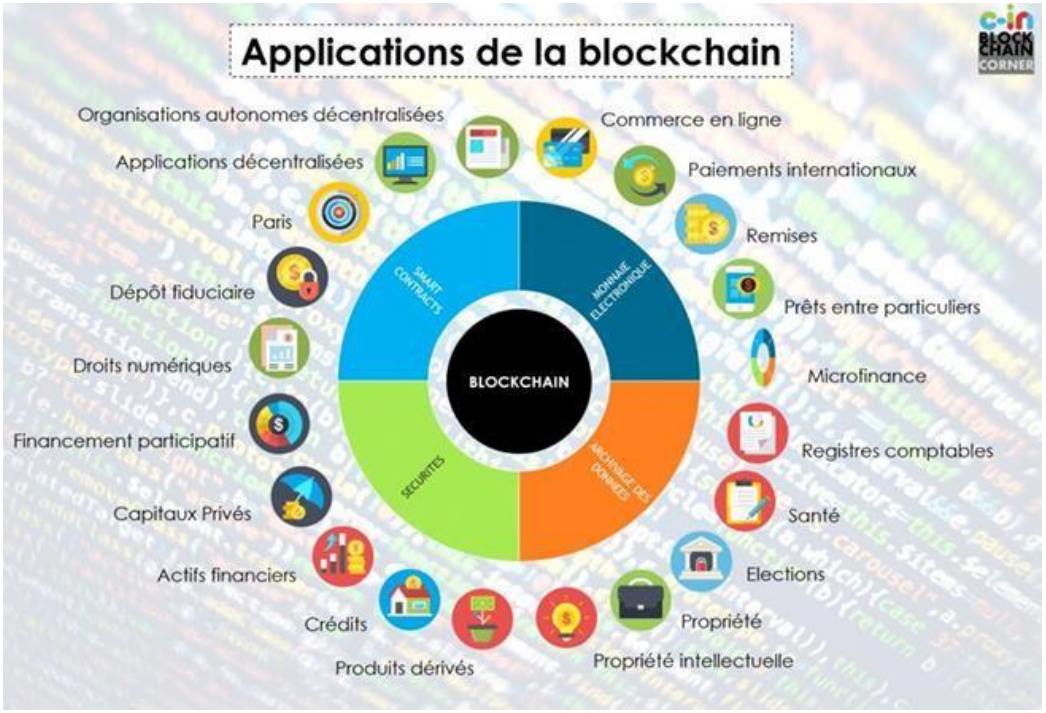

Figure 1 Cas d'utilisation de la blockchain [7]

2.3. Caractéristiques principales de la technologie Blockchain

La technologie Blockchain se caractérise principalement de six éléments majeurs : décentralisé, transparent, sécurisé et immuable, autonome, open source et anonyme. Comme décrit ci-dessus :

- La décentralisation : avec la blockchain, les informations sont distribuées sur le réseau plutôt qu'en un point central. Cela rend également le contrôle des informations à distribuer et à gérer par consensus atteint par une entrée partagée des nœuds connectés sur le réseau. Les données qui étaient auparavant concentrées en un point central sont désormais gérées par de nombreuses entités de confiance. [7]
- La transparence : la Blockchain est qualifiée d'être transparente car tout le monde peut la télécharger dans son intégralité et vérifier à tout moment son honnêteté [8].
- La sécurité : La base de données peut uniquement être étendue et les enregistrements précédents ne peuvent être modifiés (au moins, le coût est très élevé si quelqu'un souhaite modifier les enregistrements précédents). Ces enregistrements sont dits

Immuables, une fois stockés, deviennent réservés pour toujours et ne peuvent être modifiés facilement sans le contrôle simultané de plus de 51% des nœuds du réseau. Lesystème cryptographique de validation garantit qu'il est quasiment impossible de réécrire une transaction une fois son bloc validé (personne n'a réussi à le faire depuis la création du Bitcoin).

- L'autonomie : la puissance de calcul et l'espace d'hébergement sont fournis par les nœuds du réseau, c'est-à-dire les utilisateurs eux-mêmes. Il n'y a donc pas besoin d'infrastructure centrale. Au sein d'une blockchain, l'infrastructure n'est plus concentrée dans les mains d'une organisation mais est, au contraire, éclatée dans l'ensemble des points du réseau. Une blockchain est donc autoportante et indépendante de services tiers [9].
- Open source : La technologie de la blockchain est formulée de manière à fournir un accès open source à toutes les personnes connectées au réseau. Cette polyvalence inimitable permet à quiconque non seulement de vérifier publiquement les enregistrements, mais également de développer diverses applications imminentes.
- Anonymat : Lorsque le transfert de données à lieu entre nœuds, l'identité de l'individu reste anonyme, ce qui en fait un système plus sécurisé et fiable.

2.4. Types de Blockchain

La blockchain peut être "avec permission" (privée) ou "sans permission" (publique). Les blockchains avec permission restreignent les contributeurs du consensus, tandis que les blockchains publiques permettent la participation anonyme de membres illimités. Les blockchains avec permission sont économiques en termes de temps d'exécution et d'énergie, et les transactions sont généralement privées. Les blockchains publiques utilisent la cryptographie pour assurer une communication sécurisée et permettent à chaque acteur de lire, écrire et valider des transactions. En résumé, il existe trois catégories de blockchains, chacune avec ses propres caractéristiques et utilisations spécifiques.

2.4.1. Blockchains Publiques

Les réseaux de blockchain publics sont utilisés dans la plupart des monnaies numériques du marché. Chaque utilisateur crée une adresse personnelle puis interagit avec le réseau. Ils peuvent soumettre des transactions ; et peuvent ajouter des entrées au grand livre.

Cela signifie que :

1. Quiconque au monde peut vérifier l'exactitude du grand livre.
2. Même des étrangers anonymes sans autorisation explicite peuvent rejoindre le réseau et participer au processus de validation des transactions, à condition qu'ils respectent le protocole de consensus [10].

Parmi ces blockchains, on cite deux plateformes très connues :

2.4.1.1. Bitcoin

Le bitcoin est devenu la monnaie numérique cryptographique la plus célèbre depuis son introduction par Satoshi Nakamoto en 2008 et sa mise en ligne en janvier 2009 [11]. De nos jours, c'est la monnaie numérique la plus utilisée au monde et acceptée par de plus en plus de marchands en ligne. Bitcoin est une technologie paire à pair fonctionnant sans autorité centrale. La gestion des transactions et la création de bitcoins est prise en charge collectivement par le réseau. Bitcoin est libre et ouvert, sa conception est publique, personne ne le possède ni ne le contrôle, et tous peuvent s'y joindre. Grâce à plusieurs de ses propriétés uniques, cette application rend possible des usages prometteurs qui ne pourraient pas être couverts par les systèmes de paiement précédents [12].

2.4.1.2. Ethereum

En 2013, Vitalik Buterin, programmeur et cofondateur du magazine Bitcoin, a fondé Ethereum, il souhaitait un Blockchain plus volatile, qui n'était pas uniquement utilisé pour les monnaies. En 2015, il a lancé Ethereum comme une deuxième Blockchain publique, qui peut enregistrer des contrats, des emprunts, … etc.

Ethereum est un réseau de chaîne de blocs public distribué qui se concentre sur l'exécution du code de programmation de toute application décentralisée. Plus simplement, il s'agit d'une plate- forme de partage d'informations à travers le monde qui ne peut ni être manipulée ni modifiée [13].

Ethereum peut être considéré comme une Blockchain avec un langage de programmation intégré, ou comme un ordinateur globalisé, basé sur le consensus, sur lequel

les applications s'exécutent parce qu'elles valorisent les avantages offerts par Ethereum par rapport à ceux proposés par un serveur normal.

a. Compte Ethereum

Dans Ethereum, l'état est composé d'objets appelés "comptes", chaque compte ayant une adresse de 20 octets et les transitions d'état étant des transferts directs de valeur et d'informations entre comptes.
"Ether" est le principal crypto-carburant interne d'Ethereum, et est utilisé pour payer les frais de transaction. En général, il existe deux types de comptes : les comptes externes, contrôlés par des clés privées, et les comptes contractuels, contrôlés par leur code de contrat [14].

b. Transaction et Message

Dans Ethereum, nous différencions les transactions et les messages. Les "messages" sont quelque peu similaires aux "transactions" dans Bitcoin, mais avec trois différences importantes. Premièrement, un message Ethereum peut être créé soit par une entité externe, soit par un contrat, alors qu'une transaction Bitcoin ne peut être créée qu'en externe. Deuxièmement, il existe une option explicite pour que les messages Ethereum contiennent des données. Enfin, le destinataire d'un message Ethereum, s'il s'agit d'un compte contractuel, a la possibilité de retourner une réponse ; cela signifie que les messages Ethereum englobent également le concept de fonctions. Le terme "transaction" est utilisé dans Ethereum pour désigner le paquet de données signé qui stocke un message à envoyer à partir d'un compte détenu en externe [14]. La transaction est la manière dont une entité externe interagit avec Ethereum. Il peut être utilisé par un utilisateur externe pour mettre à jour l'état de l'enregistrement ou des informations stockées sur le réseau blockchain Ethereum [7].

c. La Machine Virtuelle Ethereum (EVM)

Les contrats Ethereum sont écrits dans un langage de haut niveau tel que Solidity et sont ensuite compilés en bytecode avant d'être déployés dans la blockchain via une transaction. La

machine virtuelle Ethereum est chargée d'exécuter ce code. Lors de l'exécution, le code est stocké soit sur la pile, soit en mémoire, soit dans le stockage du compte du contrat. Les deux premiers seront nettoyés après la fin du processus.

L'exécution du code fait partie de la validation des transactions. Le code est donc exécuté par tous les nœuds qui valident les transactions. L'état du système, qui contient tous les comptes, est stocké dans les blocs [7].

2.4.2. Blockchains privées

Une blockchain privée est considérée comme un réseau centralisé puisqu'il est entièrement contrôlé par une seule organisation [16]. Une blockchain privée peut donc être considérée comme un constructeur de confiance dans un réseau sans confiance et le catalyseur d'un marché ouvert désintermédié, préservant la confidentialité [10]. Dans la blockchain privée, son accès et son utilisation sont limités à certains acteurs. Personne ne peut y participer sans y être autorisé mais tout le monde peut la consulter [18]. Certaines propriétés de la blockchain privée telles que le consensus, le registre distribué, le journal transparent et la communication P2P, et les contrats intelligents rendent ce type de blockchain adapté aux organisations financières et aux banques [17]. Parmi les plus célèbres, citons Hyperledger (de Linux Foundation) et Ripple (protocole permettant les transferts internationaux).

2.4.2.1. Hyperledger

Le projet Hyperledger est une initiative open source de la Fondation Linux lancée en 2015 pour permettre une collaboration intersectorielle dans le développement d'applications blockchain et des outils qui les entourent. L'objectif est de répondre aux limites d'évolutivité et de fiabilité des solutions blockchain existantes afin de faciliter leur adoption pour les applications industrielles.

D'après [43] la technologie basée sur Hyperledger fonctionne en utilisant ces couches :

- Une couche de consensus, qui conclut un accord sur la commande et confirme si les transactions d'un bloc sont correctes.

- Une couche de contrat intelligent, qui traite et autorise les demandes de transaction.
- Une couche de communication, qui gère le transport des messages peer-to-peer (P2P).
- Une API, qui permet à d'autres applications de communiquer avec la blockchain.
- Des Services de gestion d'identité, qui valident les identités des utilisateurs et des systèmes.

2.4.2.2. Ripple

C'est une technologie de paiement basée sur une plateforme de registre distribué qui permet aux utilisateurs de transférer des devises numériques, des actifs numériques et des informations. Développé par Ripple Labs Inc. (anciennement connu sous le nom d'Open Coin Inc.) et lancé en 2012. Selon [3], la plateforme Ripple permet des transactions en temps réel avec des frais très bas et une grande scalabilité. Il est utilisé par de nombreuses institutions financières à travers le monde, notamment Santander, American Express, Standard Chartered et UBS.

2.4.3. Blockchains consortiums (hybride)

Le système de blockchain de consortium est généralement considéré comme une fusion de blockchain publique et privée. Dans la blockchain de consortium, un groupe d'organisations ou d'individus est chargé de prendre des décisions concernant la validation des blocs et le consensus [17]. Ce type de blockchain est souvent utilisé dans les secteurs très réglementés tel le secteur bancaire, comme le consortium R3 qui regroupe plus de 70 institutions financières dans le monde [19]. Seul un groupe de nœuds présélectionnés participerait au processus de consensus d'une Blockchain du consortium.

Propreté	**Blockchain Publique**	**Blockchain Consortium**	**Blockchain Privée**
Public blockchain	Tous les mineurs	Ensemble de nœuds sélectionné	Une organisation
Read permission	Publique	Peut-être public ou restreint	Peut-être public ou restreint
Immutabilité	Presque impossible à falsifier	Peut-être trafiqué	Peut-être trafiqué
Efficience	Bas	Haute	Haute
Centralisé	Non	partiel	Oui

Consensus processus	Sans autorisation	Autorisée	Autorisée

Tableau 2 Comparaisons entre la blockchain publique, la blockchain de consortium et la blockchain privée

2.5. Fonctionnement

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

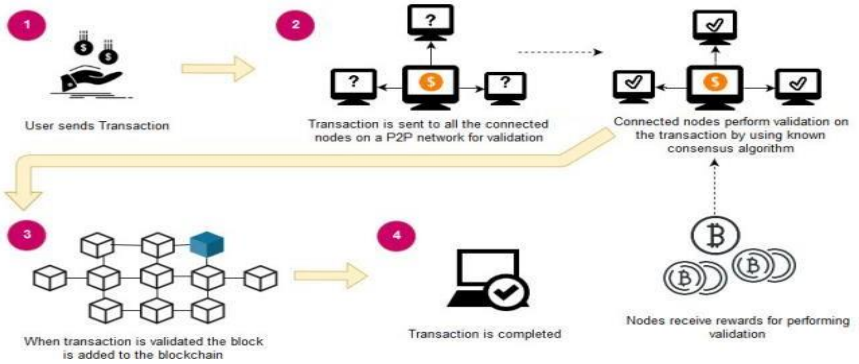

Figure 2 Un aperçu de l'architecture de la blockchain. [7]

1. Lorsqu'un utilisateur envoie une nouvelle transaction sur le réseau blockchain, un nouveau bloc est créé pour y stocker cette transaction. Ce bloc est ensuite diffusé à tous les nœuds connectés du réseau, qui vérifient son intégrité et l'ajoutent à leur propre copie de la blockchain.
2. Le processus d'ajout de blocs sur la blockchain est réalisé par les nœuds qui parviennent à un consensus sur les blocs valides à ajouter. Ces nœuds utilisent des algorithmes pour vérifier les transactions et s'assurer de l'authenticité de l'expéditeur. Les nœuds qui parviennent à valider un bloc sont récompensés par une crypto-monnaie. Ce processus de validation est appelé minage et les nœuds qui effectuent cette tâche sont appelés mineurs.
3. Une fois la validation effectuée, le bloc est ajouté à la blockchain, ce qui contribue à l'enregistrement et à la vérifiabilité des transactions précédentes.

Une fois l'ensemble du processus de validation effectué, la transaction est terminée.

2.6. Architecture de Blockchain

La Blockchain est une combinaison d'ordinateurs reliés les uns aux autres au lieu d'un serveur central, ce qui signifie que tout le réseau est décentralisé [21].

Voici les principaux composants de l'architecture blockchain :

2.6.1. Nœud

Les nœuds blockchain sont des dispositifs utilisés par les parties prenantes pour exécuter le logiciel de protocole d'un réseau décentralisé. Leur rôle est de maintenir le consensus, valider les transactions et assurer la sécurité du grand livre public de la blockchain [21].

10 types de nœuds blockchain on cite les plus importants [20] :

Nœud Complet : Serveurs d'un réseau décentralisé qui conservent l'historique complet des transactions, synchronisent, stockent, copient et distribuent les données tout en validant les nouveaux blocs.

Nœud d'autorité : Les nœuds d'autorité sont élus par une communauté pour agir en tant que modérateurs d'une blockchain privée ou partiellement centralisée.

Nœud Miniers : Incités par les crypto-monnaies fraîchement frappées, les nœuds de minage vérifient les transactions à l'aide d'un modèle de consensus de preuve de travail, une méthode de validation qui repose sur des énigmes cryptographiques arbitraires, afin de débloquer des jetons et d'ajouter de nouveaux blocs à une blockchain. Les mineurs sont des ordinateurs, travaillant généralement en groupe, qui sont la propriété d'une entité, comme un individu ou une entreprise.

Nœud léger : Traite rapidement les transactions et dépend des nœuds complets pour fonctionner, ne téléchargeant pas la blockchain complète.

2.6.2. Transaction

Dans la blockchain, une transaction est une valeur de transfert qui est diffusée sur le réseau et transmise à tous les participants du réseau. Une transaction est toute opération qui consiste à modifier l'état de la blockchain en ajoutant des données qui seront stockées de façon irréversible. Elle est annoncée dans le réseau P2P et reçue par les nœuds validateurs [10]. Les transactions peuvent décrire un transfert d'actifs (crypto-monnaie ou autre), ou une interaction avec un contrat intelligent.

2.6.3. Bloc

Bloc est une structure de données utilisée pour conserver un ensemble de transactions qui est distribué à tous les nœuds du réseau. [20] Les blocs sont reliés entre eux pour former une chaîne. Le premier bloc est appelé bloc de genèse ou bloc 0. Les blocs sont identifiés soit par le hachage de leur en-tête, soit par leur hauteur, c'est-à-dire la distance qui les sépare du bloc de genèse. La chaîne est créée en incluant le hachage du bloc n-1 dans l'en-tête du bloc n. [22]

Un bloc contient trois éléments qui sont des données, le hachage du bloc actuel et le hachage du bloc précédent. Les données peuvent être n'importe quoi car cela dépend du type de blockchain. [7]

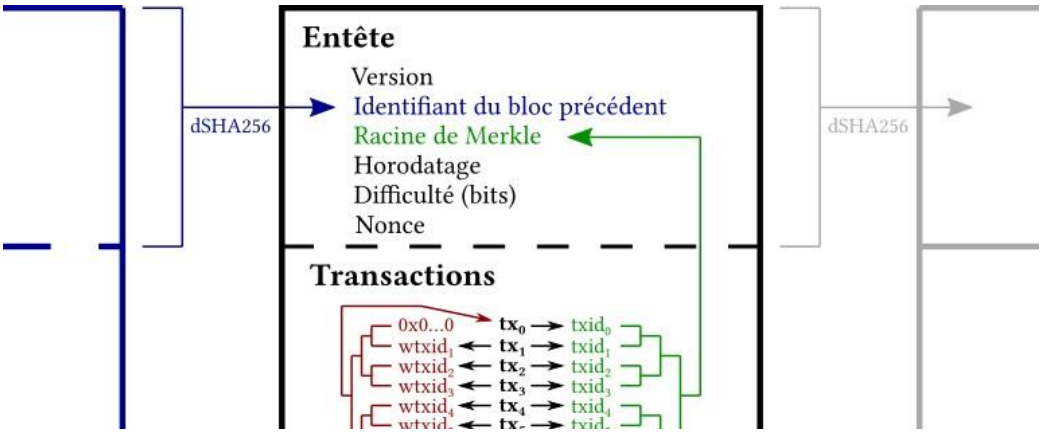

Figure 3 Structure d'une Blockchain [24]

- Le champ racine Merkel fait référence à la racine d'un arbre à distorsion qui stocke les informations de transaction dans chaque bloc.
- Horodatage : est un élément de données stocké dans chaque bloc, indiquant le moment exact de l'extraction et de la validation du bloc [24].
- La difficulté est une mesure de la façon de trouver un hachage inférieur à une cible donnée. Une difficulté élevée signifie qu'il faudra plus de puissance de calcul pour extraire le même nombre de blocs, ce qui rend le réseau plus sûr contre les attaques.
- 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.

2.6.4. Minage

Le minage est le processus d'ajout de transactions [18].

2.6.5. Contrat intelligent

Également appelés (contrats blockchain, contrats numériques ou contrats auto-exécutables). Le nom '' Smart Contract '' remonte à l'année 1996, au cours de laquelle Szabo a proposé un contrat intelligent basé sur un ensemble de promesses et de protocole afin d'éliminer interventions de tiers [25]. Le contrat intelligent a un objectif similaire à un contrat normal dans le monde physique. Il diffère en ce sens qu'il s'agit d'une forme numérique et qu'il est stocké dans le système de blockchain. Cependant, la technologie blockchain a rendu possible ce concept de contrat intelligent/privé en raison de sa nature décentralisée. Dans la blockchain basée sur les contrats intelligents, les détails de la transaction ne sont pas stockés sur des blocs, mais un contrat intelligent est écrit qui contient toutes les données et informations liées à la transaction. Le contrat intelligent est comme un code programmable fonctionnant sur la blockchain que les nœuds IoT peuvent écrire en fonction des exigences de la transaction, puis ils peuvent exécuter le contrat dans le réseau blockchain [26]. Une fois que le contrat est déployé dans la blockchain, il commence à être exécuté et aucun utilisateur IoT ne peut arrêter cette exécution, pas même le créateur du code [17].

2.6.6. Hachage

Le hachage fait essentiellement référence au processus qui prend une entrée de longueur variable et donne une sortie de longueur fixe. Cette sortie de longueur fixe est appelée un hache. Le traçage d'un hachage facilite le suivi d'une transaction. La fonction de base d'un hachage est de vérifier l'intégrité du bloc et de former un maillon de chaîne en incluant le bloc de hachage précédent dans l'en-tête du bloc actuel. Si un autre nœud altère le bloc, la valeur de hachage change, provoquant une incompatibilité de valeur de hachage et rendant la chaîne locale invalide [25].

Une fonction de hachage cryptographique doit posséder certaines propriétés pour être considérée comme sécurisée.

- Déterministe : la même entrée donnera toujours le même hachage.
- Calcul rapide : La valeur de hachage d'un message se calcule très rapidement.
- Résistante pré-image : il est impossible de déterminer l'entrée à partir d'un hachage donné.
- De petits changements dans l'entrée modifient le hachage : Même si vous apportez un petit changement dans votre entrée, les changements qui seront reflétés dans le hachage seront énormes.
- Résistant aux collisions : Étant donné deux entrées différentes A et B où H (A) et H (B) sont leurs hachages respectifs, il est impossible que H(A) soit égal à H(B). Cela signifie que pour la plupart, chaque entrée aura son propre hachage unique.

2.6.7. Consensus

Dans la blockchain, le consensus entre des nœuds non fiables est un défi similaire au problème des généraux byzantins, qui a été soulevé dans [28]. Dans ce problème, un groupe de généraux doit coordonner leur attaque sur une ville, mais des traîtres peuvent falsifier les messages, rendant la communication difficile. De même, dans la blockchain, où le réseau est distribué, il n'y a pas de nœud central garantissant la cohérence des registres. Atteindre un consensus dans un tel environnement est essentiel pour assurer la cohérence des données. Différentes approches sont utilisées pour atteindre le consensus dans la blockchain [29]. Ces protocoles garantissent que les différents nœuds ont des grands livres cohérents, assurant ainsi

l'intégrité du système.

a. PoW (Proof of Work)

Le consensus dans le réseau Bitcoin est atteint grâce à une stratégie appelée Preuve de Travail (PoW) [11]. Dans cette méthode, les mineurs, représentant 51% du réseau, sont choisis pour enregistrer les transactions en effectuant des calculs informatiques. Cela garantit la sécurité du réseau en rendant difficile les attaques à plus de 51%. Les mineurs calculent des valeurs de hachage en modifiant le nonce dans l'en-tête du bloc, cherchant une valeur inférieure à une cible donnée. Lorsqu'un mineur trouve une valeur de hachage valide, il diffuse le bloc aux autres nœuds pour confirmation. Une fois validé, le bloc est ajouté aux blockchains des autres mineurs. Cependant, le minage PoW peut être énergivore, et certains protocoles comme Prime coin [30] ont été conçus pour utiliser les calculs dans des applications secondaires, comme la recherche mathématique, afin de réduire le gaspillage de ressources.

b. PoS (Proof of stake)

Le protocole de preuve d'enjeu (PoS) est une alternative économe en énergie au PoW basé sur des sanctions plutôt que des récompenses pour la protection [31]. Contrairement au PoW où tous les participants peuvent agir en tant que mineurs, dans le PoS seuls les validateurs qui ont verrouillé leur capital peuvent valider les transactions et créer de nouveaux blocs [32]. Les validateurs sont choisis en fonction de la taille de leur mise, et leur chance d'être sélectionnés est proportionnelle à leur mise. Cela évite la centralisation indésirable basée sur le solde du compte. Les validateurs proposent et valident les nouveaux blocs en votant, et la décision du prochain validateur est prise en fonction du poids du vote de chaque validateur. Le PoS vise à maintenir la sécurité et la décentralisation tout en réduisant la consommation d'énergie.

c. BFT (Byzantine Fault Tolerance)

La tolérance aux pannes byzantines (BFT) est essentielle dans les systèmes de consensus

pour parvenir à un accord malgré des comportements malveillants ou défaillants [33]. Les systèmes de blockchain décentralisés gèrent la blockchain en tant que grand livre mondial distribué sans autorité centrale. Les algorithmes de consensus tels que PoW et PoS sont utilisés pour prendre en charge les transactions, mais ne sont pas idéaux pour la tolérance aux pannes byzantines. Pour éviter ces défauts, de nouveaux protocoles et algorithmes doivent être développés. Comprendre le BFT est important pour utiliser la blockchain dans divers domaines, comme les soins de santé, où des ressources informatiques importantes ne sont pas pratiques. Dépendre du vote des nœuds tout en évitant le PoW pourrait être une solution viable pour résoudre les problèmes de fautes byzantines.

2.7. Sécurité et défis dans la blockchain

La blockchain est une technologie émergente qui se répand dans divers secteurs et qui présente un grand nombre d'avantages et d'opportunités. Cependant, cette technologie présente son propre ensemble de défis à relever. Quelques-uns de ces défis majeurs sont abordés dans cette section.

2.7.1. Problème de sécurité et de confidentialité

La technologie blockchain présente des problèmes de sécurité et de confidentialité. Les mécanismes de consensus et de confirmation des transactions peuvent être manipulés par des acteurs contrôlant plus de 50% des nœuds, ce qui constitue une attaque à 51%. Les blockchains sont également vulnérables aux pertes de données et aux interruptions du réseau si les transactions ne sont pas surveillées de près. Les attaques sybiles, où des nœuds malveillants créent de nombreuses identités pour tromper le réseau, sont une menace [34], [35]. Les attaques DDoS et le vol de messages sont également fréquents sur les réseaux blockchain, visant les services monétaires tels que le minage et les échanges de crypto-monnaies [36]. Le minage égoïste est une pratique où un groupe de mineurs retient délibérément des blocs pour maximiser ses profits. Le stockage de données de santé sur une blockchain peut entraîner des

retards dans les transactions, des fuites de données et la divulgation d'informations sensibles des patients. Malgré ses avantages potentiels, la sécurité et la confidentialité restent des préoccupations majeures dans la mise en œuvre actuelle de la technologie blockchain [37].

2.7.2. Limites d'évolutivité

Les systèmes de blockchain existants présentent des problèmes de débit, d'efficacité et de coût de calcul. Les limitations de la taille des blocs entraînent des temps de traitement longs, ce qui dégrade l'efficacité du système et rend le grand livre distribué encombré [38],[39]. La quantité massive de données générées par les appareils IoT rend difficile le traitement de ces ensembles de données par une blockchain. En raison de ces inconvénients, de nombreux développeurs d'applications ne considèrent pas la technologie blockchain comme un remplacement viable dans la gestion des réseaux IoT complexes [37]. De plus, le coût élevé de calcul, notamment lié aux frais d'exécution des transactions, limite l'adoption de la blockchain. Les mécanismes de consensus tels que la preuve de travail nécessitent une quantité considérable de puissance de calcul, ce qui pose des problèmes pour les dispositifs IoT aux ressources limitées. La complexité d'un système de blockchain exige des ressources technologiques et humaines importantes, ce qui suscite des préoccupations concernant les coûts de maintenance élevés et entrave l'utilisation généralisée des services basés sur la blockchain [37].

2. Contrôles d'accès

2.1. Définition des contrôles d'accès

Le contrôle d'accès est un élément essentiel de toute organisation et essentiel à la sécurité des systèmes informatiques. L'accès à ces systèmes et la manipulation de leurs données sont contrôlés de manière appropriée en fonction des niveaux de classification des données décrits dans les politiques de contrôle d'accès (ACP) [40].

Le contrôle d'accès a été un élément majeur dans l'application des exigences de sécurité et de confidentialité des informations et des ressources en ce qui concerne l'accès non autorisé [41].

Le contrôle d'accès est une technique de sécurité qui régule qui ou quoi peut voir ou utiliser des ressources dans un environnement informatique. Il s'agit d'un concept fondamental de la sécurité qui minimise les risques pour l'entreprise ou l'organisation [42].

2.2. Définition d'une politique de contrôle d'accès

Une politique de contrôle d'accès est une définition de la manière dont un système doit accorder ou refuser l'accès, qui peut aller d'une déclaration abstraite telle que "seuls les utilisateurs figurant sur cette liste doivent avoir accès" ou "seuls les utilisateurs qui m'ont rendu service dans le passé doivent avoir accès", à des langages de politique avec une sémantique exécutable (opérationnelle) [43].

A partir de cette définition nous distinguons trois concepts fondamentaux d'une politique de contrôle d'accès :

- Sujet : c'est une entité active qui représente les utilisateurs dans un système. Le sujet est généralement une personne, une application, un processus, une adresse IP …etc.
- Objet : entité passive qui représente les données à protéger contenues dans le système. L'objet peut être un fichier, ressource, table relationnelle, programme, information …etc.
- Action : représente l'opération possible appelée par le sujet sur l'objet. L'action peut être lire, écrire, exécuter, modifier, supprimer …etc.

2.3. Modèle de contrôle d'accès à base d'attribut (ABAC)

Le modèle ABAC pour (Attribute Based Access Control), a été développé par L. Wang, D. Wijesekera, S. Jajodia [44], propose de définir les droits d'accès en se basant sur des caractéristiques liées à la sécurité de chaque entité appelés attributs. De ce fait, les droits d'accès à une ressource ou un service sont définis pour un ou plusieurs attributs que les identités sont susceptibles de posséder. Généralement On distingue trois groupes d'attributs, selon le type de l'entité à laquelle ils s'appliquent :

- 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.
- 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 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.
- Les attributs d'environnement : Le modèle ABAC prend en compte le contexte d'exécution du système en définissant des attributs d'environnement tels que la date, le niveau de sécurité du réseau, le débit de la connexion, etc.

Ce paradigme offre donc plus de flexibilité [45]. De plus, en définissant un attribut se rapprochant de la notion de rôle, ABAC permet de simuler le comportement d'un modèle RBAC. Il permet notamment de déterminer des droits d'accès avec une granularité plus fine. De plus, en définissant un rôle comme un ensemble d'attributs, il est plus facile de gérer les conflits. Par ailleurs, la gestion des droits d'accès est facilitée, car elle ne nécessite pas d'informations supplémentaires.

Ce modèle nous semble approprié pour l'expressivité de la spécification des politiques d'autorisation [45]. Il donne les moyens d'écrire des politiques, intégrant une diversité d'informations de sécurité et associées à des exigences propres aux organisations. Il est considéré comme une solution de gestion de politiques d'autorisation de manière générique.

Enfin l'ABAC introduit la gestion du contexte, via les entités d'environnements. Ceci permet d'obtenir des modèles plus souples, qui peuvent s'adapter à différentes situations et de dynamiser ainsi la prise de décision pour le contrôle d'accès.

2.4. Formalisme des politiques de sécurité ABAC

Le modèle ABAC peut être décomposé en deux aspects : le modèle de politique, qui

définit la formalisation d'une politique de sécurité et le modèle d'architecture, qui définit la façon dont les autorisations sont vérifiées dans le système. Voici la syntaxe définie par ABAC pour exprimer des politiques de sécurité [46].

S, R et E sont respectivement les sujets, les ressources et les environnements.

SAK1 ≤ k ≤ K, RAm(1 ≤ m ≤ M), AEn(1 ≤ n ≤ N) sont respectivement les (k-ième, m- ième et n-ième) attributs d'un sujet, d'une ressource, d'un environnement (avec k, m et n compris entre 1 et le nombre d'attribut définit pour chaque entité) ;

ATTR(s), ATTR(r) et ATTR(e) sont les relations d'attributions des attributs aux entités (sujet, ressource et environnement) respectivement.

On a donc par exemple ATTR(s) ⊆ SA1 x SA2 x ... x SAK.

Les attributs sont définis de la façon suivante : ATTR(s)=valeur.

Exemple : CurrentDate(e) = 6-12-2008 signifie qu'on affecte la valeur 6-12-2008 à l'attribut d'environnement CurrentDate ;

Les règles sont définies comme étant des fonctions booléennes des attributs de s, r et e :

Rule: can_access(s, r, e) ← f (ATTR(s), ATTR(r), ATTR(s))

On définit ensuite une base de règles de politiques comme un ensemble de règles regroupant plusieurs sujets et plusieurs ressources au sein d'un même domaine de sécurité. La gestion des autorisations se fera alors via l'évaluation de l'ensemble des règles de la base de politiques.

Exemples concrets

Soit un ensemble de sujets et de ressources. On définit pour chaque sujet un attribut nommé « Role » et pour chaque ressource un attribut Name. La règle « Les managers peuvent accéder aux ressources nommées ApprovePurchase » s'exprime alors de la façon suivante :

Rule 1 : can_access (s, r, e) ← (Role (s) = 'Manager') ∧ (Name(r) = 'ApprovePurchase')

Par exemple, pour imposer qu'une ressource puisse seulement être consultée par ses propriétaires, On définit un attribut UserID désignant l'identifiant d'un sujet, un attribut ResourceOwner désignant le propriétaire de la ressource et nous pouvons avoir une règle

comme suit :

R2: can_access (s, r, e) ← (UserID(s) = ResourceOwner(r))

2.5. Architecture d'autorisation

Une fois les politiques définies, il est nécessaire d'expliciter la façon dont le système va effectuer les vérifications des règles avant de fournir ou non l'autorisation d'accès. L'architecture d'autorisation définie par ABAC est la suivante [46] :

- Les AA (Atributes Authorities) : sont les entités responsables de la création et de la gestion des attributs. Ils sont également responsables d'établir les relations entre les attributs, leur valeur et l'entité correspondante.
- Le PEP (Policy Enfoncement Point): est l'entité chargée d'effectuer les requêtes d'autorisation et d'appliquer la politique. Il est logiquement situé entre les sujets et les ressources, ce qui lui permet d'intercepter toute tentative d'accès, d'effectuer la requête vers le système de sécurité afin de vérifier si la tentative d'accès est autorisée ou non. Notons que s'il est représenté ici comme un point unique, il peut être physiquement distribué en plusieurs points du système. La seule condition étant que le système soit architecturée de manière à ce qu'il ne soit pas possible d'accéder à une ressource protégée sans passer par le PEP.
- Le PDP (Policy Decision Point): est l'entité chargée d'évaluer les politiques applicables et de prendre une décision concernant une requête d'accès à une ressource par un sujet. Il reçoit donc les requêtes du PEP, contacte les différents AA pour récupérer les attributs qui ne sont pas présents dans la requête, et applique les règles de sécurité pour donner sa décision au PEP (accès autorisé ou refusé).
- La Policy Authority : crée et gère les politiques de contrôle d'accès (règles de décision, conditions, etc.).

2.6. Avantage du ABAC

- Introduction de la notion de gestion de contexte via les entités d'environnement. Ceci permet d'obtenir des modèles plus souples, qui peuvent s'adapter à différentes situations et de dynamiser ainsi la prise de décision pour le contrôle d'accès. Cela est très appréciable pour l'application sur des SOA (Services Orienté Architectures), ou de nombreux paramètres peuvent influer sur la disponibilité de certaines ressources.
- Un modèle beaucoup plus adapté à la principale problématique liée au contrôle d'accès au sein de SOA, qui était le dynamisme d'accès aux informations.
- L'utilisation des attributs offre une granularité très fine pour définir les règles [19] d'autorisation: Il suffit de définir un attribut pour prendre en compte un nouveau paramètre entrant en jeu dans la définition des autorisations, qu'il s'applique aux sujets, aux ressources ou aux environnements.

3. L'Advanced Encryption Standard

L'AES est un algorithme de chiffrement symétrique basé sur le système Rijndael construit par Joan Daemen et Vincent Rijmen [58]. Il permet de chiffrer et déchiffrer l'information par blocs de 128 bits (16 octets binaires), à l'aide de clés de 128, 192 ou 256 bits. L'AES s'exécute en plusieurs tours qui sont composés de plusieurs transformations [58]. Le nombre de tours Nr dépend de la taille des clés où 10 tours sont nécessaires pour des clés de 128 bits, 12 tours pour des clés de 192 bits et 14 tours pour des clés de 256 bits [59].

3.1. Chiffrement et déchiffrement avec l'AES

Dans ce qui suit, nous détaillons l'AES-128, où les 128 bits de données sont répartis en 16 blocs de 8 bits (8 bits = 1 octet), eux-mêmes « dispatchés » dans un tableau 4×4. Même les 128 bits de la clé sont organisés sous forme matricielle [BAB 12]. Pour d'évidentes raisons de taille, les valeurs binaires seront notées sous forme hexadécimale.

La première étape de chiffrement consiste à combiner la matrice State (le bloc de texte clair) avec la clé. Cette opération s'appelle AddRound Key. À chaque tour, quatre

transformations sont appliquées SubBytes, ShiftRows, MixColumn et AddRoundKey sauf pour le dernier tour, l'opération MixColumns n'est pas effectuée. Chaque tour utilise sa propre sousclé qui est générée par

l'opération Key Expansion à partir de la clé maitresse [59].

Le déchiffrement est l'opération inverse du chiffrement et les transformations se réalisent dans le sens inverse.

La figure montre les différentes opérations effectuées dans chaque round pendant le processus de chiffrement et de déchiffrement de l'AES.

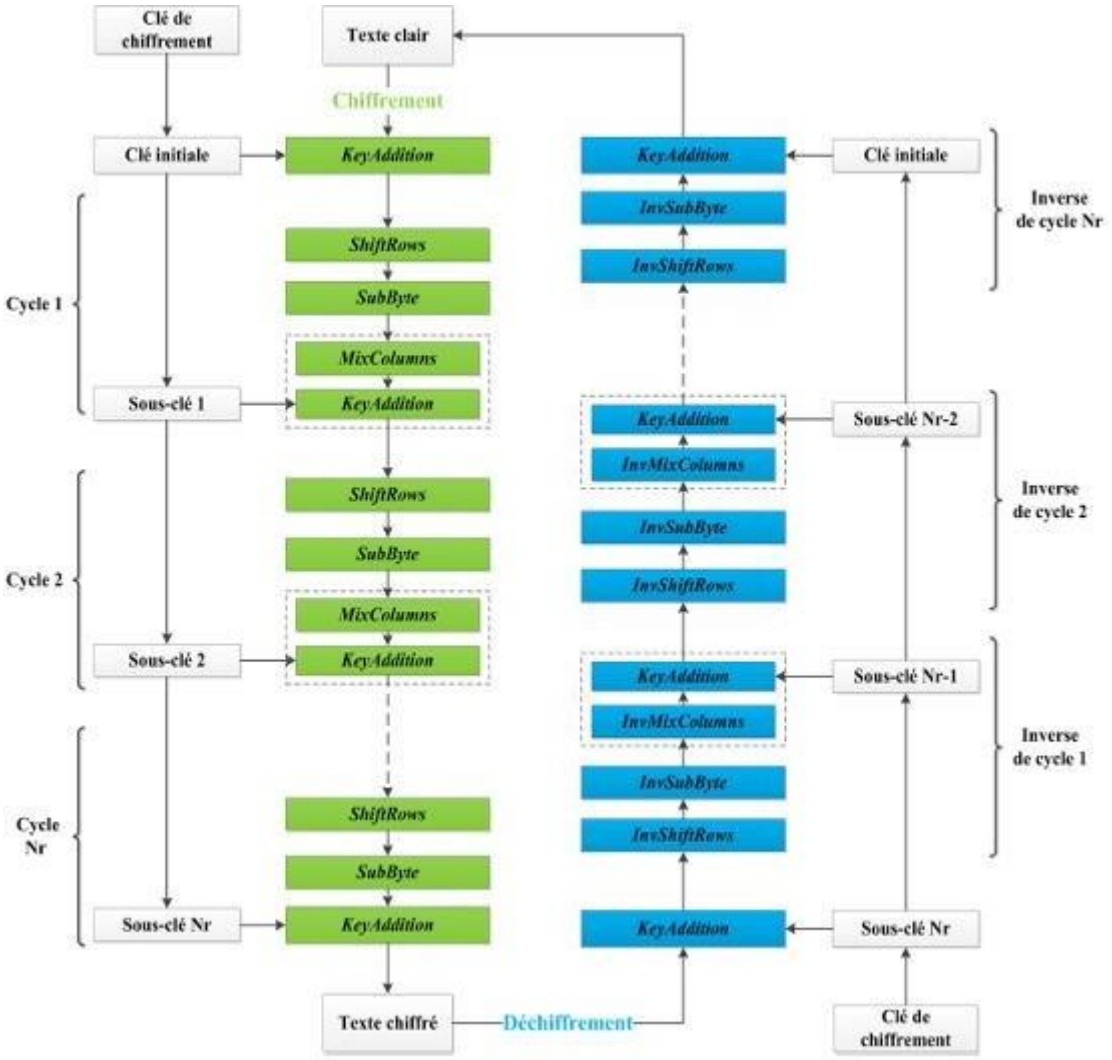

Figure 4 Transformations de l'AES [60]

- SubBytes : Chaque octet de l'état est remplacé par un autre octet à l'aide d'une table de substitution appelée S-Box.

- ShiftRows : Les octets de chaque ligne de l'état sont décalés circulairement vers la gauche. La première ligne reste inchangée, la deuxième ligne est décalée d'une position, la troisième ligne de deux positions et la quatrième ligne de trois positions.

- MixColumns : Les colonnes de l'état sont mélangées en utilisant des opérations Mathématiques spécifiques. Cela permet de confondre les relations entre les octets.

-AddRoundKey : L'état est combiné avec une clé de tour en utilisant l'opération XOR (ou exclusif). Cela introduit l'élément de clé dans l'état.

3.2. Caractéristiques et points forts de l'AES

Le choix de cet algorithme répond à de nombreux critères plus généraux dont nous pouvons citer les suivants [61] :

- Sécurité ou l'effort requis pour une éventuelle cryptanalyse.
- Facilité de calcul : cela entraine une grande rapidité de traitement
- Besoins en ressources et mémoire très faibles
- Flexibilité d'implémentation : cela inclut une grande variété de plateformes et d'applications ainsi que des tailles de clés et de blocs supplémentaires.
- Simplicité : le design de l'AES est relativement simple.

4. E-santé

4.1. Les Hôpitaux intelligents « Smart Hospital »

Un hôpital intelligent utilise des technologies innovantes pour améliorer les soins aux patients et optimiser la gestion des ressources. Il fonctionne de manière ingénieuse et créative, en évaluant continuellement ses processus opérationnels pour faire face aux contraintes financières et aux réductions de personnel. La conception des installations, y compris les infrastructures et les salles, est également essentielle pour une gestion efficace des patients, des fournitures et des appareils.

4.2. Définition

En tant que concept similaire aux hôpitaux intelligents, le terme « hôpitaux intelligents » a commencé à être utilisé dans le contexte de l'identification par radiofréquence (RFID), dans le consortium de soins de santé parrainé par la RFID Éducationnel Fondation aux États-Unis en 2009. Avec ce concept, les chercheurs ont proposé un service qui applique la technologie de localisation en temps réel, la technologie de communication et la technologie d'interopérabilité à divers espaces dans les hôpitaux tels que les salles d'opération, les services hospitaliers et les

cliniques externes [68].

La littérature a introduit les concepts d'hôpitaux intelligents de diverses manières, comme la montre ci-dessus :

- ENISA définit un hôpital intelligent comme : Un hôpital qui améliore le processus de traitement des patients basé sur l'IoT, optimise la gestion des actifs en établissant un environnement TIC connecté aux actifs internes de l'hôpital et utilise un processus d'automatisation commercial [69].
- KESSIA l'a défini comme : Un établissement médical qui a mis en place un système de gestion intégré pour des soins sûrs aux patients et une gestion hospitalière efficace en utilisant les TIC pour diverses ressources appartenant à l'hôpital, telles que le personnel médical, les installations, les informations et l'équipement [70].

4.3. Dossier Médical Electronique

Le DME (Dossier Médical Électronique) est un outil de stockage sécurisé des données des patients sous forme numérique, permettant la continuité, l'efficacité et la qualité des soins. Il fournit une vue historique, courante et prospective de l'état de santé du patient, facilitant l'aide à la décision et la planification des soins pour les professionnels de santé [54].

L'ISO donne également un certain nombre d'autres termes couramment utilisés pour décrire différents types de DME[58].

- Dossier médical électronique (Electronic medical record (DME)) : généralement axé sur les soins médicaux.
- Dossier patient électronique (Electronic patient record (EPR)) : Contient toutes ou la plupart des informations cliniques du patient d'un hôpital particulier.
- Dossier patient informatisé (Computerized patient record (CPR)) : contient la totalité ou la plupart des informations cliniques du patient d'un hôpital particulier
- Dossier de santé électronique (Electronic Heath care record (EHCR)) : Contient toutes les informations sur la santé du patient.

- Dossier de santé personnel (Personal Heath record) : Contrôlé par le patient et contenant des informations au moins en partie saisies par le patient.
- Dossier médical informatisé (Computerized medical record) : Créé par numérisation d'image d'un dossier de santé papier.
- Dossier médical numérique (Digital medical record) : un dossier en ligne tenu par un fournisseur de soins de santé.
- Dépôt de données cliniques (Clinical data repository) : Un magasin de données opérationnelles qui contient et gère les données cliniques recueillies auprès des fournisseurs de services de santé.
- Dossier client électronique (Electronic client record) : La portée est définie par les professionnels de la santé autres que les médecins, p. par des physiothérapeutes ou des travailleurs sociaux.
- DSE virtuel (Virtual EHR) : aucune définition faisant autorité.

4.4. Les avantages de DME

Les DME offrent une accessibilité facile et rapide aux informations médicales, permettant à plusieurs utilisateurs autorisés de les consulter simultanément à distance. Cela améliore la qualité des informations en éliminant les limitations telles que les données manquantes ou illisibles, et permet aux médecins d'accéder et de mettre à jour les dossiers des patients à partir de différents endroits via Internet ou d'autres systèmes de télécommunication.

Les DME présentent également des avantages de productivité. Les médecins pourraient accéder aux informations concernant les antécédents de la maladie du patient avant la visite du patient. La recherche d'informations sous une forme informatisée est devenue plus facile et simplifiée, puisque les enregistrements peuvent être recherchés et affichés selon différents paramètres [55].

4.5. Exigences de sécurité des applications

Les exigences de sécurité et de confidentialité pour les applications e-santé sont difficiles, telles que l'authentification mutuelle, l'anonymat de l'utilisateur, la non-traçabilité, le secret de transmission parfait, l'accord de clé de session et la résistance aux attaques pour assurer la confidentialité et la sécurité des données [56].

a. Authentification mutuelle

Cela peut être réalisé en utilisant des protocoles d'authentification tels que l'authentification Kerberos. Nous avons analysé à partir de la littérature que la sécurité de la couche de transport (TLS)/couche de socket sécurisée (SSL) assure le flux de communication mais ne peut pas vérifier l'utilisateur ou le dispositif de communication, qui peut être vérifié par authentification mutuelle. Il permet uniquement à l'utilisateur autorisé d'accéder aux informations du serveur [56].

b. Anonymat

Si un attaquant obtient l'identité de l'utilisateur, la vie privée du patient peut être compromise et ce n'est pas pratique pour les personnes âgées. Par conséquent, l'anonymat est l'une des exigences de sécurité. L'identité du patient et 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 [56].

c. Non traçabilité

Si un attaquant retrace les exercices de communication de clients spécifiques, il peut alors deviner la véritable identité des patients avec une probabilité plus élevée. Il en résulte une violation de la vie privée des utilisateurs. Un attaquant ne peut pas décider des exercices de

communication d'un utilisateur spécifique [56].

d. Secret de transmission parfait

Le secret de transmission parfait (PFS) est utilisé pour l'accord de clé, qui protège les sessions précédentes contre l'accord futur de mots de passe ou de clés privées en créant une clé de session pour chaque session [57]. Ici, un attaquant ne peut pas accéder aux clés de session, qui ont été créées lors de sessions antérieures ; même si n'importe qui peut accéder à la clé privée de l'utilisateur, cela ne peut pas affecter car la clé de session est chiffrée avec plusieurs algorithmes cryptographiques [56].

e. Résistance aux attaques

Cela signifie riposter contre l'utilisateur qui a attaqué le système. Il résiste à divers types d'attaques telles que la relecture, le MIM, l'attaque par usurpation d'identité, l'attaque par usurpation d'identité et l'attaque par modification [56].

5. Conclusion

En conclusion, ce premier chapitre a mis en évidence l'importance de la blockchain, du contrôle d'accès et des dossiers médicaux électroniques dans divers secteurs, en particulier la santé. La blockchain offre une technologie de base de données distribuée sécurisée, tandis que le contrôle d'accès garantit que seuls les utilisateurs autorisés peuvent accéder aux données. L'AES est utilisé de manière efficace pour chiffrer ces dossiers sensibles, assurant ainsi la confidentialité des informations. En gardant un œil sur les développements futurs, nous pourrons continuer à améliorer la sécurité et la confidentialité des dossiers médicaux électroniques, contribuant ainsi à l'amélioration des soins de santé et à la protection des informations sensibles des patients.

ChapitreII: Conception

1. Introduction

Dans cette partie, nous allons décrire la conception de notre solution pour la gestion de données médicales à l'aide de la technologie blockchain et le contrôle d'accès ABAC avec l'algorithme de chiffrement AES. Nous allons aborder les différents aspects de la conception, en commençant par la description générale de notre solution avec la caractéristique nécessaire de la blockchain et les composants de notre architecture, puis en détaillant les différentes fonctionnalités. Nous allons également expliquer comment nous avons utilisé les smart contrats pour stocker les données médicales de manière sécurisée et transparente. Enfin, nous allons présenter l'étude conceptuelle de notre solution.

2. L'objectif du travail

La blockchain est une technologie de stockage et de transmission d'informations, qui permet de stocker des données de manière décentralisée et sécurisée, en utilisant un réseau de nœuds connectés qui valident les transactions. Chaque bloc de données dans la blockchain est sécurisé à l'aide de cryptographie et est lié de manière chronologique aux blocs précédents, formant ainsi une chaîne de blocs. Cette technologie offre un haut niveau de sécurité et de transparence, car chaque transaction est vérifiée et validée une autorité de confiance, ce qui rend difficile la falsification ou la suppression de données. La blockchain est utilisée dans de nombreux domaines, y compris la finance, la logistique, l'assurance, l'immobilier, et maintenant également dans le domaine de la santé pour la gestion des données médicales.

L'objectif principal de ce projet est de fournir une plateforme de gestion des données médicales en utilisant la Blockchain et le contrôle d'accès par attribut ABAC et réalisé par le chiffrement AES qui soit à la fois sécurisée, efficace et rentable, permettant aux professionnels de la santé d'avoir un accès rapide et facile aux informations médicales pertinentes, tout en garantissant la confidentialité et la sécurité des données des patients. Cette plateforme permettra également de réduire les coûts administratifs liés à la gestion des données médicales et d'améliorer la collaboration entre les différents services de soins de santé.

3. 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 assure le plus haut niveau de sécurité[].

- Dans un premier temps, nous avons essayé de trouver la meilleure utilisation de la technologie blockchain dans le domaine des applications e-santé. À cette fin, nous avons exploré les chaînes de blocs publiques et si elles peuvent être utilisées tout en garantissant la sécurité des informations. Après de nombreuses recherches, nous avons conclu que l'utilisation d'une blockchain privée, telle que décrite dans le chapitre 1, serait la meilleure solution pour assurer le plus haut degré de sécurité et de contrôle sur les informations.

- En outre, il fallait contrôler l'accès aux données médicales de patient pour cela, la plateforme utilise le contrôle d'accès basé sur les attributs avec le chiffrement AES pour garantir que seuls les utilisateurs autorisés ont accès aux informations confidentielles des patients. Les autorisations d'accès sont basées sur des politiques de sécurité prédéfinies qui sont définies en fonction des rôles et des niveaux de permission des utilisateurs.

- La plateforme comprend également des fonctionnalités de traçabilité et d'audit pour permettre la vérification de l'intégrité et de l'authenticité des données. Chaque transaction enregistrée sur la blockchain est associée à une signature numérique unique, permettant une traçabilité aisée des données depuis leur origine jusqu'à leur stockage sur la blockchain. Cette fonctionnalité est particulièrement utile lorsqu'il s'agit de données sensibles, telles que des dossiers médicaux, car elle permet de vérifier l'origine et l'authenticité des données à tout moment. Les professionnels de la santé peuvent accéder aux informations médicales pertinentes de manière sécurisée et privée.

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

4. Caractéristique de notre Blockchain

Pour notre application, nous avons choisi une blockchain privée. Les blockchains privées sont plus adaptées aux applications d'e-santé car elles offrent une plus grande confidentialité et sécurité des données et permettent de contrôler l'accès aux données par des entités spécifiques telles que les hôpitaux et les sociétés médicales. En effet, seules les personnes autorisées (médecins ou patients) peuvent accéder et gérer leurs propres données et dossiers médicaux.

- Parce que nous avons choisi une plate-forme privée pour le système, les nœuds du système sont connus et validés par les gestionnaires du personnel médical et les gestionnaires de patients. Par conséquent, le nombre de nœuds décentralisés est faible par rapport au Bitcoin et il n'est pas nécessaire de mettre en place un mécanisme PoW très gourmand (en termes de consommation d'énergie). Deuxièmement, nous préférons déployer des mécanismes PoW plus légers tels que la preuve d'autorité PoA", car ils ne nécessitent pas de calculs complexes ou de concurrence pour valider les transactions. Cela en fait une option attrayante pour les réseaux privés où la vitesse et l'efficacité sont primordiales.

- Les nœuds sont généralement contrôlés par des entités impliquées dans le réseau (par exemple, des hôpitaux, des centres de santé, etc. et dans notre cas un administrateur). Les nœuds peuvent être hébergés sur des serveurs locaux. Autrement dit, ils peuvent être hébergés dans les locaux de ces entités plutôt que sur des serveurs distants. Cela permettra une gestion plus directe et sécurisée des données de santé par ces établissements.

- Les transactions peuvent être :

➢ Transaction de lecture : qui ne nécessitent pas de consensus car elles ne modifient pas les données stockées sur la blockchain. Il permet uniquement de lire les données existantes de la blockchain. Cela signifie qu'ils peuvent être exécutés rapidement et efficacement. Par exemple :

- Authentification d'utilisateurs.
- Lecture des dossiers médicaux.
- Lecture des attributs du médecin.

➢ Transaction d'écriture : Un consensus ou une autorité de confiance est nécessaire pour valider ces transactions. Cela dépend de la sensibilité et de l'importance des données traitées par le système. Exemple de ces transactions :

- Création d'un nouvel utilisateur.
- Modification des attributs d'un médecin.
- Création/modification d'un dossier.

➢ Nous stockons des informations d'identification qui aident à vérifier l'identité de l'utilisateur effectuant la transaction. La blockchain stocke également les attributs et les données nécessaires au cryptage AES utilisé pour protéger la confidentialité des données des patients. Enfin, il stocke de manière sécurisée et immuable les dossiers des patients, y compris les informations médicales pertinentes, sur la blockchain.

5. Contrôle d'accès basé sur les attributs et sur le chiffrement AES

Dans notre système, l'utilisateur possède un ou plusieurs attributs qui sont gérés par l'autorité de confiance. Le patient à un attribut de valeur dynamique qui diffère d'un patient à un autre, par exemple le numéro d'identification unique pour le droit d'accès d'un patient à son dossier médical.

Le médecin a des attributs qui dépendent de leurs Spécialité médicale, Lieu d'exercice c.-à-d. dossiers des patients de son propre établissement ou de son propre service, Niveau de responsabilité, ou une Dans notre système, nous avons mis en place un système de contrôle d'accès basé sur les attributs pour assurer la confidentialité des dossiers médicaux. Ce système permet d'accorder des autorisations spéciales à certains utilisateurs, tels que les médecins spécialisés dans des domaines particuliers ou les patients qui souhaitent partager leurs données avec des médecins spécifiques.

Chaque utilisateur se voit attribuer des attributs spécifiques qui déterminent leur accès aux dossiers médicaux. Par exemple, un médecin spécialiste en cardiologie peut se voir attribuer l'attribut "cardiologue", lui permettant ainsi d'accéder aux dossiers médicaux des patients liés

à cette spécialité.

Pour garantir la confidentialité des données stockées dans la blockchain, nous utilisons l'algorithme de chiffrement AES (Advanced Encryption Standard). Lorsque les données sont stockées, elles sont chiffrées à l'aide de l'AES en utilisant une clé de chiffrement. Cela signifie que seuls les médecins qui ont les attributs requis par les patients appropriés peuvent décrypter et accéder aux informations.

Cette approche assure la confidentialité des dossiers médicaux tout en permettant un accès contrôlé aux utilisateurs autorisés. Les autorisations peuvent être modifiées ou révoquées à tout moment, offrant ainsi une flexibilité dans la gestion des accès aux dossiers médicaux.

En adoptant l'algorithme AES et en utilisant un système de contrôle d'accès basé sur les attributs, nous garantissons la confidentialité des données sensibles tout en permettant aux professionnels de la santé d'accéder aux informations pertinentes pour assurer des soins appropriés.

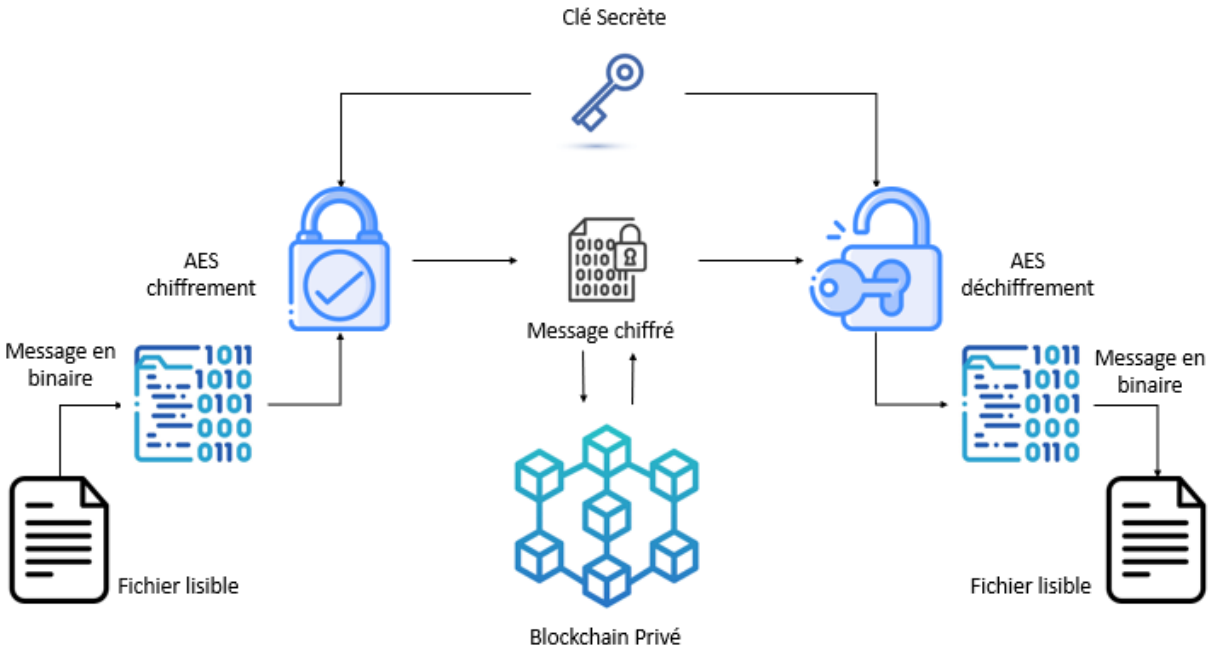

Figure 5 l'algorithme de chiffrement et conversion du fichier médicale

6. Architecture du système

L'architecture proposé de notre système se compose de :

●L'autorité de confiance (Administrateur) : Fait l'organisation du système, l'administrateur

peut ajouter, supprimer ou mettre à jour les nœuds dans le système, il peut vérifier l'identité des utilisateurs à partir d'une base de données externe et les valider pour qu'ils puissent stocker leurs informations dans la blockchain. L'administrateur peut également gérer les attributs. En outre, il peut surveiller et maintenir le bon fonctionnement du système, gérer les mises à jour de sécurité et assurer la conformité réglementaire du système.

●Les patients : peuvent contrôler l'accès à leur dossier médical. Il crée des droits d'accès pour les professionnels de la santé tels que les médecins afin qu'ils puissent consulter les informations médicales pertinentes. Les patients peuvent également consulter leur propre dossier médical pour vérifier les informations qui y sont stockées.

●Médecin : qui ajoute les documents de suivi au dossier médical du patient si leurs attributs correspondent à la politique d'accès définie par le patient. Un médecin peut également consulter les dossiers médicaux des patients auxquels il a été autorisé à accéder. En outre, les médecins peuvent également générer des rapports sur la santé des patients basés sur des données stockées dans des dossiers médicaux en fonction des droits d'accès et des politiques définis par le patient.

●Réseaux blockchain privé : pour stocker les clés de chiffrement et les attributs utilisateur, en plus les données médicales.

●DAPP : pour permettre aux utilisateurs d'interagir avec l'application.

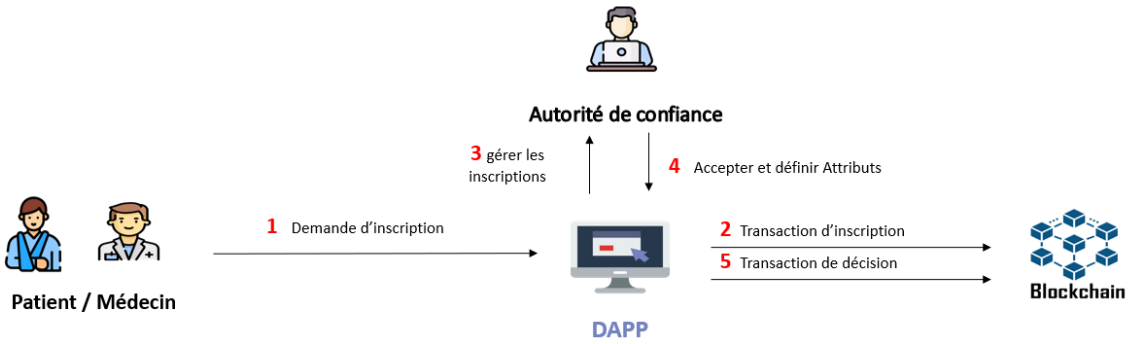

Figure 6 architechture du systeme

6.1. Gestion de contrôle d'accès à base d'attributs

Les hachages sont cruciaux dans les systèmes de blockchain. Dans une blockchain, chaque bloc contient un hachage représentant les transactions de ce bloc et une référence au hachage

du bloc précédent. Cette structure de chaîne garantit l'intégrité et la sécurité des données stockées dans la blockchain. Dans le cas de la blockchain Ethereum, la fonction de hachage Keccak-256 est utilisée pour calculer les empreintes.

6.2. Hachage Keccak-256 (SHA-3)

Keccak est une famille de fonctions de hachage cryptographique basées sur la construction en éponge qui utilise comme bloc de construction une permutation à partir d'un ensemble de 7 permutations [1]. Elle prend un message de taille arbitraire en entrée et renvoie un résumé de taille fixe de 256 bits. Cette fonctionnalité est conçue pour résister aux attaques cryptographiques telles que les collisions, les pré-images et les secondes pré-images. Il est considéré comme une fonction de hachage sécurisée et est largement utilisé dans les protocoles de sécurité pour garantir l'intégrité des données et la confidentialité des informations.

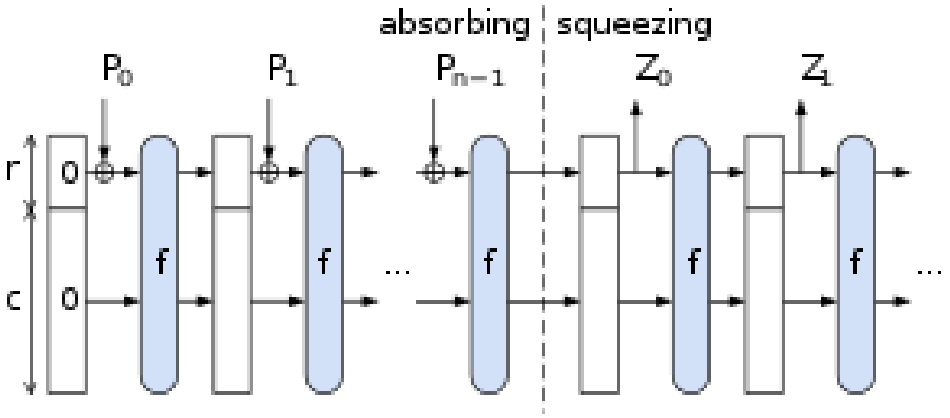

Figure 7 Construction d'une éponge pour les fonctions de hachage

Voici comment fonctionne la fonction de hachage Keccak-256 :

- **Préparation :** Le message d'entrée est d'abord rempli avec un rembourrage de sorte que salongueur soit un multiple de 1088 bits, la taille du bloc de traitement.
- **Absorption :** Les messages d'entrée sont divisés en morceaux de 1088 bits, qui sont ensuite traités les uns après les autres par un processus appelé "absorption".

Lors de l'ingestion, chaque bloc est combiné avec l'état interne de la fonction de hachage, qui est initialement défini sur une valeur constante.

- Squeezing : Une fois que tous les blocs ont été absorbés, la fonction de hachage passe en mode "squeeze". Cela signifie générer un bloc de sortie de 256 bits à partir de l'état interne de la fonction. Ces blocs initiaux sont concaténés pour former l'empreinte finale.
- **Résultat :** l'empreinte résultante de la fonction de hachage est une valeur de 256

Etude conceptuelle de notre solution

7. Etude conceptuelle de notre solution

Nous allons créer une application qui doit répondre aux spécifications fonctionnelles suivantes afin de mettre en œuvre notre solution :

- Gestion des dossiers médicaux : Les professionnels de santé doivent pouvoir créer et gérer les dossiers médicaux de leurs patients, y ajouter des fiches de suivi et les visualiser.
- Chiffrement et déchiffrement symétrique AES : Les données des dossiers médicaux doivent être chiffrées à l'aide d'AES et déchiffrées uniquement par les utilisateurs autorisés.
- Stockage des données dans le Blockchain : Les dossiers médicaux doivent être stockés dans un Blockchain privé Ethereum pour garantir leur immutabilité et sécurité.
- Extraction des données de la Blockchain : Les utilisateurs autorisés doivent pouvoir extraire les données des dossiers médicaux depuis le Blockchain.
- Partage des données entre les utilisateurs autorisés : Les utilisateurs autorisés doivent pouvoir partager les données des dossiers médicaux avec d'autres utilisateurs autorisés de manière sécurisée et contrôlée.

7.1. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation globale est un outil utile pour représenter les interactions entre les acteurs et le système, ainsi que pour identifier les fonctionnalités importantes du système. Dans le cadre de notre application, nous

avons identifié les acteurs suivants :

- Le personnel de l'hôpital : ils ont accès aux dossiers médicaux des patients.
- Les patients : peuvent accéder à leurs dossiers et gérer les autorisations d'accès.
- L'administrateur : il est responsable de la gestion des demandes d'inscription et des attributions des médecins.

Ces acteurs peuvent utiliser plusieurs fonctionnalités dans le système, telles que celles illustrées ci-dessous :

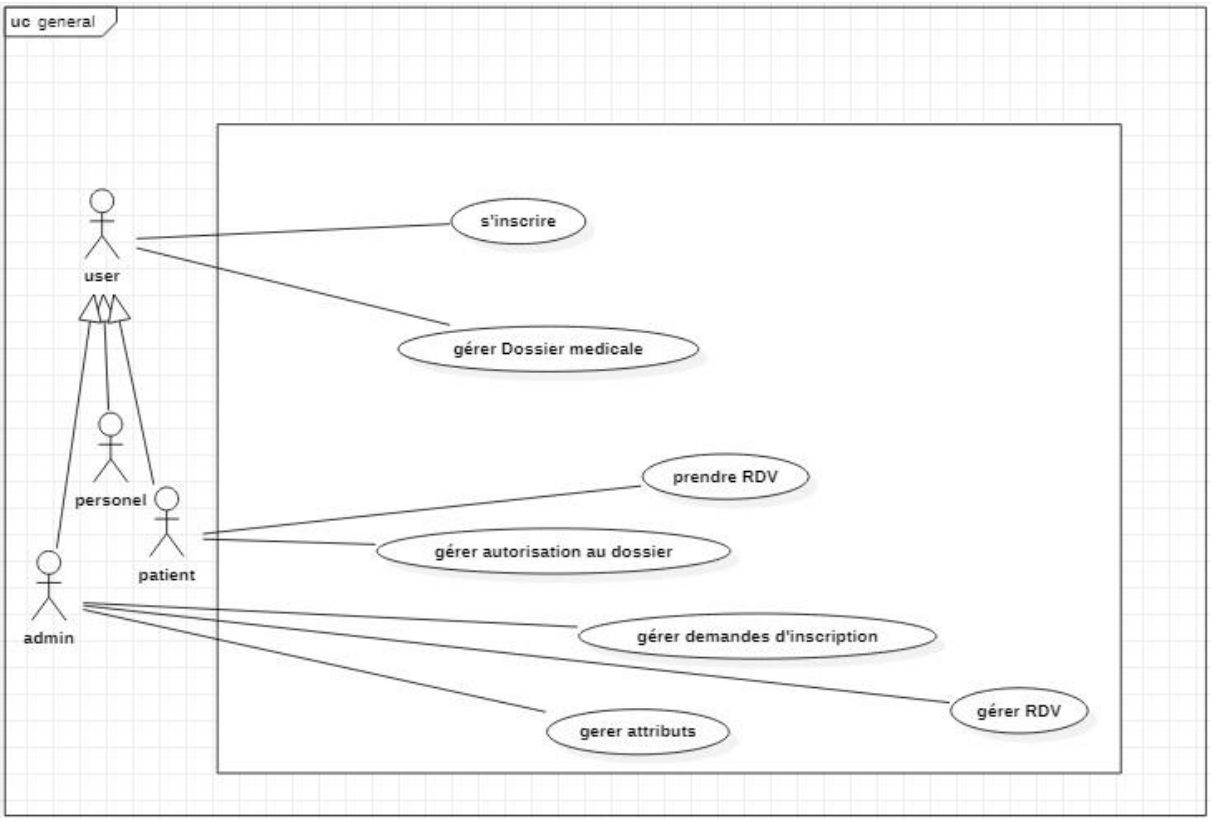

Figure 8 Diagramme de cas d'utilisation globale

Cas d'utilisation	Acteurs	Description
Inscription	Patient, personnel, admin	Les acteurs peuvent s'inscrire sur la plateforme.
Authentification	Patient, personnel, admin	Doivent s'authentifier pour accéder à l'application.

Gérer dossier médical	patient, personnel	les acteurs peuvent rechercher, consulter les dossiers. Les médecins peuvent ajouter des fiches de suivis.
Gérer l'autorisation au dossier	patient	Le patient peut créer des politiques d'accès pour son dossier.
Gérer les demandes d'inscriptions	admin	l'admin qui accepte ou refuse toute demande d'inscription à la plateforme.

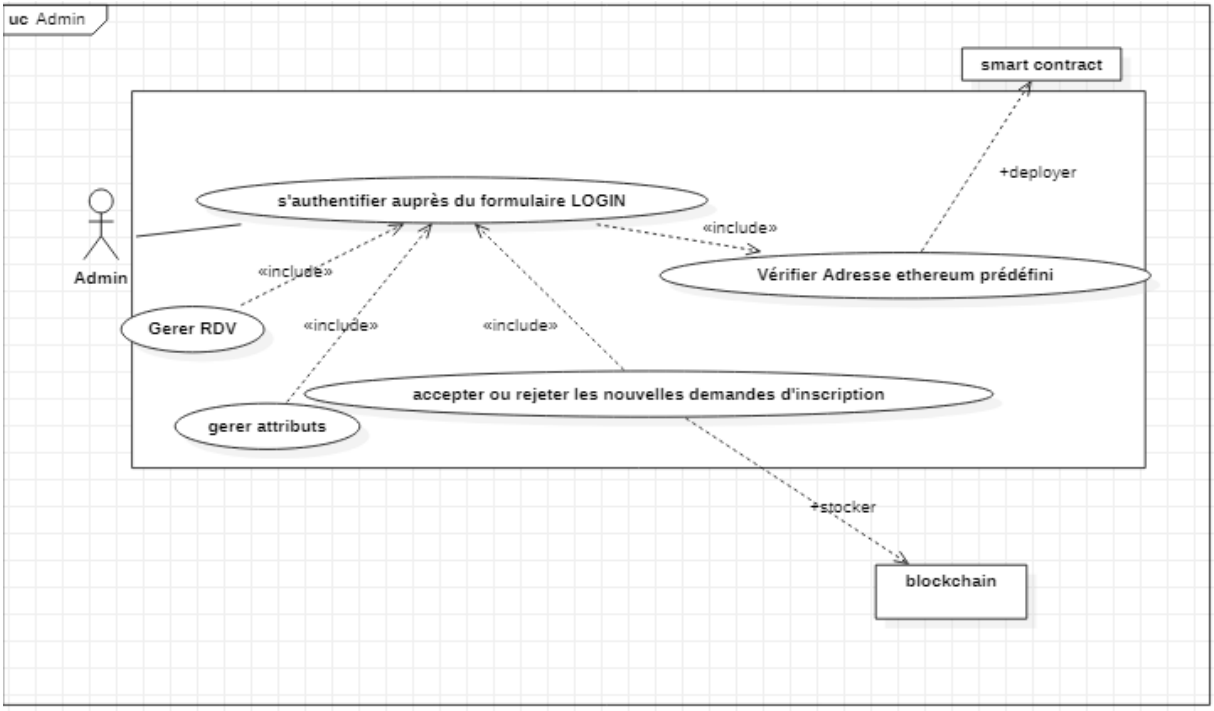

Figure 9 Diagramme de cas d'utilisation authentification admin

Dans ce diagramme, on présente les tâches attribuées à l'administrateur, telles que la gestiondes rendez-vous et des demandes d'inscription du personnel, ainsi que la gestion de leurs attributs. Cependant, avant de pouvoir effectuer ces tâches, l'administrateur doit s'authentifier dans le système.

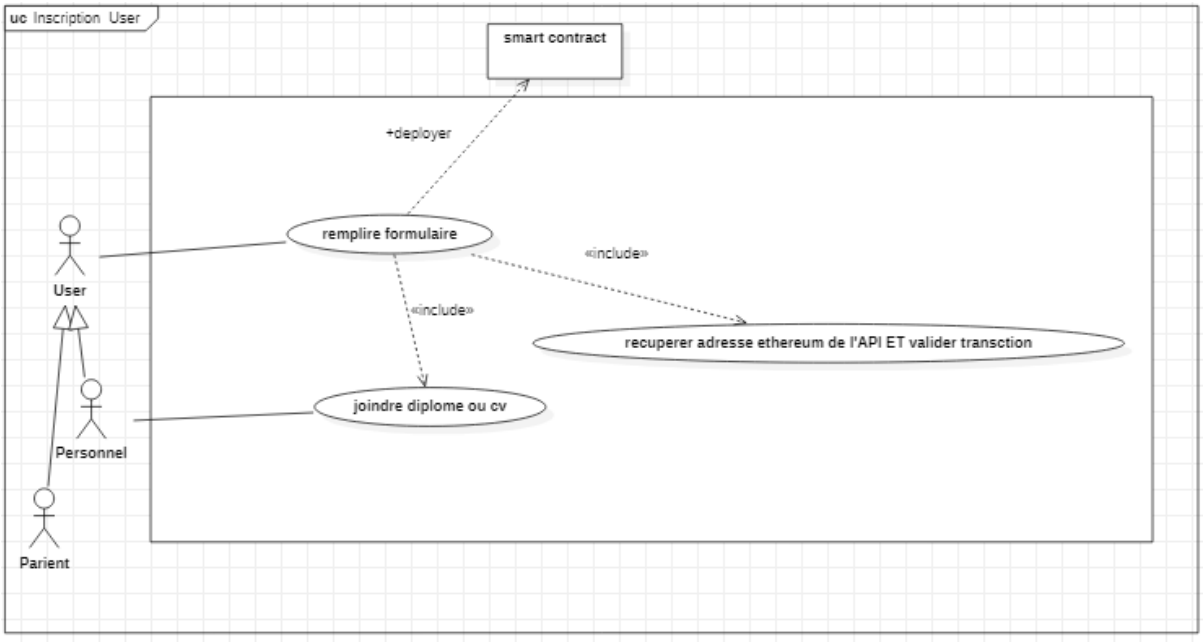

Figure 10 Diagramme de cas d'utilisation inscription utilisateur

Dans ce diagramme, nous illustrons le processus d'inscription des utilisateurs dans notre application. Tout d'abord, les utilisateurs doivent remplir le formulaire d'inscription en fournissant leur adresse Ethereum à partir de l'API. Ensuite, L'administrateur vérifie les documents fournis par l'utilisateur pour prouver qu'il s'agit

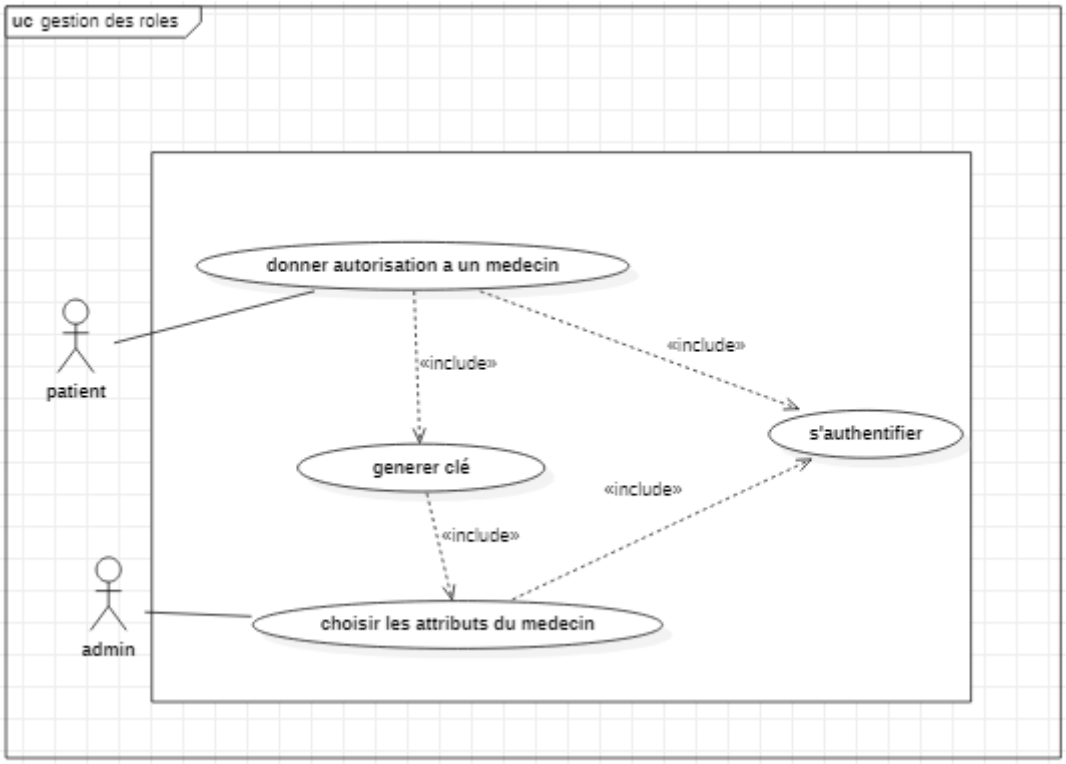

Figure 11 Diagramme de cas d'utilisation gestion des rôles

bien d'un professionnel de santé.

Dans ce diagramme, nous illustrons le processus d'attribution des rôles, où l'administrateur définit les attributs d'un personnel de santé. Ensuite, Le processus se poursuit avec le patient qui choisit un médecin et autorise l'accès à son dossier. Le système vérifie les attributs du médecin et déchiffre les données sensibles du patient en utilisant la clé secrète.

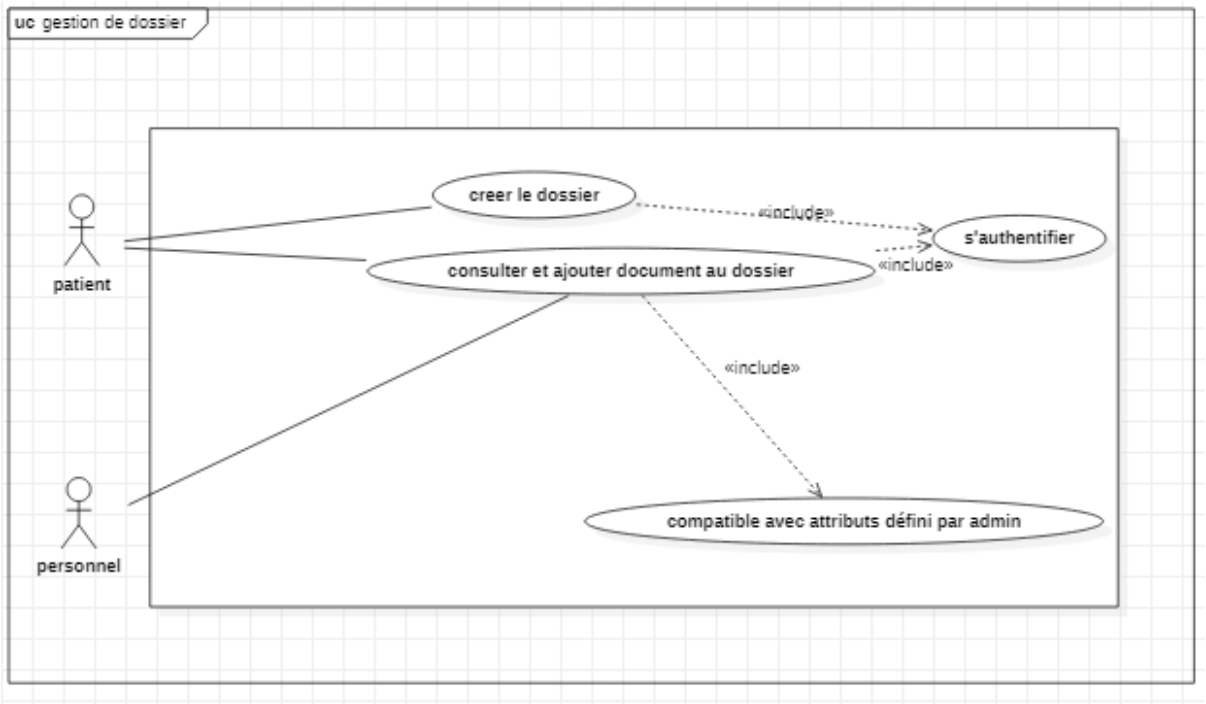

Figure 12 Diagramme de cas d'utilisation gestion de dossier médical

Le diagramme de cas d'utilisation montre comment les utilisateurs peuvent consulter ou ajouter des documents au dossier médical d'un patient, sous réserve de satisfaire les attributs prédéfinis par l'administrateur.

7.2. Diagrammes de séquence

Nous allons dans cette section décrire le fonctionnement de quelques fonctions principales de notre application.

a. Inscription des utilisateurs

Afin d'interagir avec notre application, les utilisateurs (patients ou professionnels de santé) doivent s'inscrire au préalable. Donc, une fois que l'utilisateur a fourni ses informations, celles-ci sont envoyées à l'admin pour validation. Si l'autorité de confiance confirme l'identité de l'utilisateur, un certificat lui est attribué pour confirmer son inscription. Ensuite, l'utilisateur peut utiliser ce certificat pour accéder à la plateforme et utiliser les fonctionnalités de l'application.

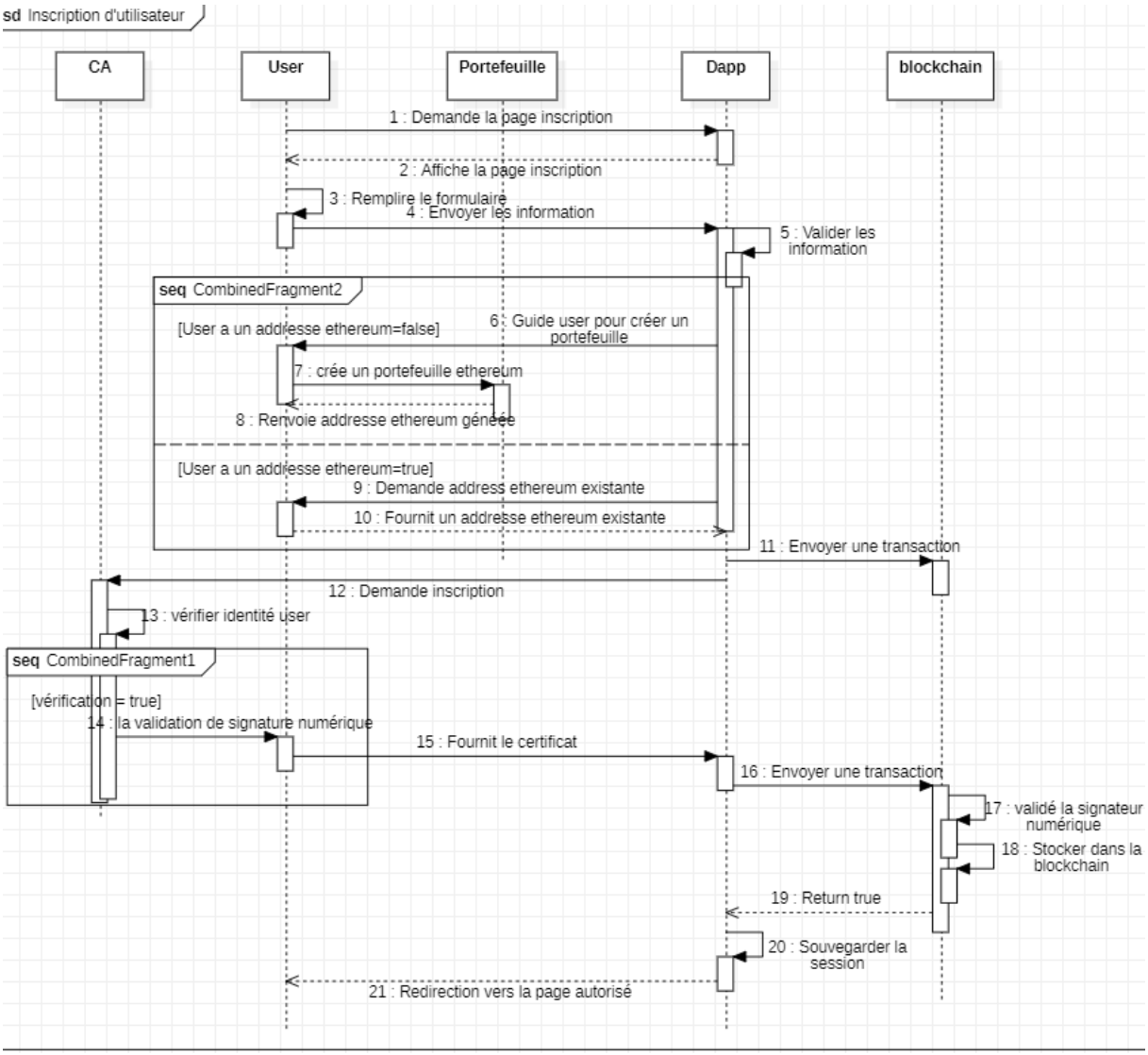

Figure 13 Diagramme de séquence inscription d'un utilisateur

b. Authentification d'un utilisateur

Avant de pouvoir utiliser les fonctionnalités de notre application, les utilisateurs doivent s'authentifier pour accéder à celle-ci.

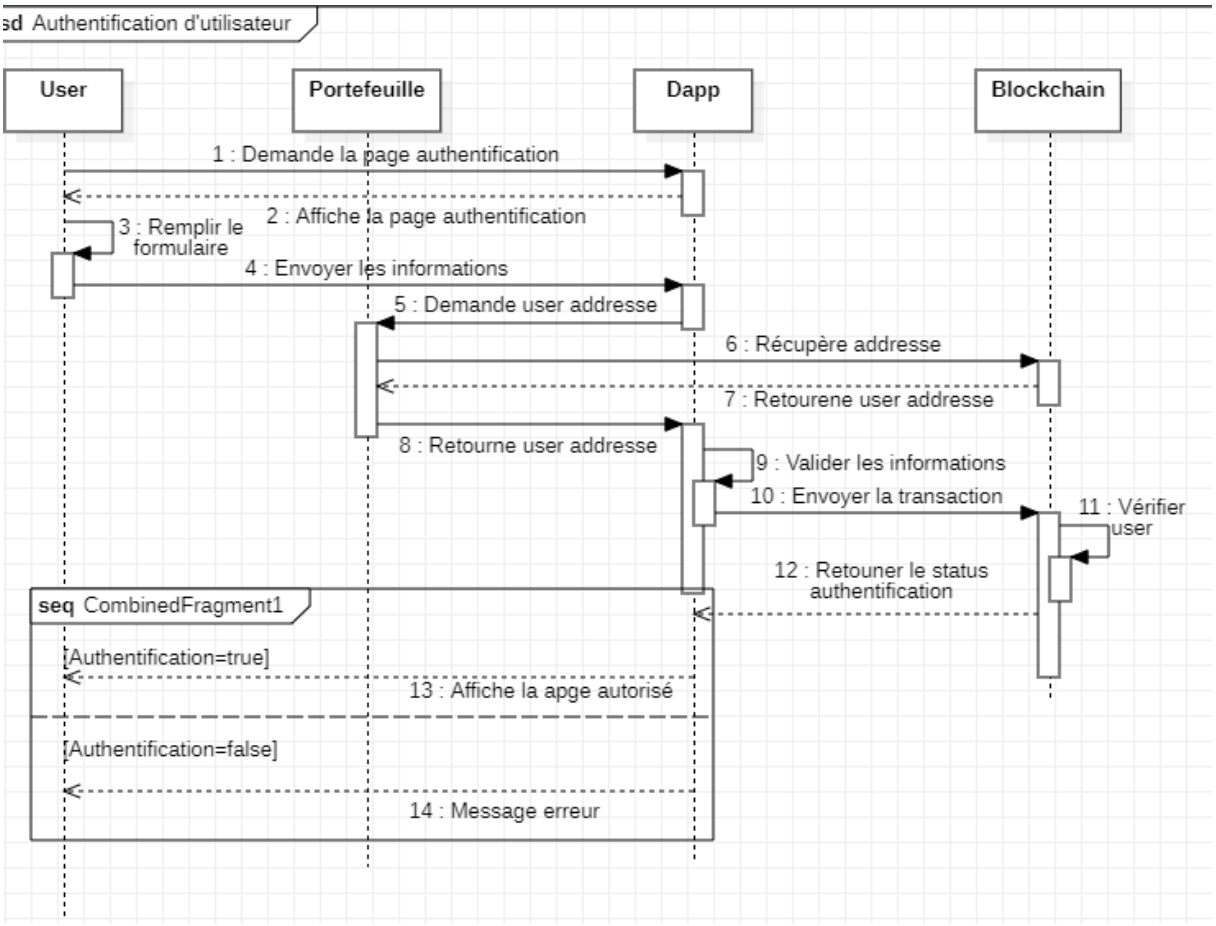

Figure 14 Diagramme de séquence authentification d'un utilisateur

c. Gestion des attributs des utilisateurs

Dans notre application, nous avons mis en place un contrôle d'accès à base d'attributs. L'autorité deconfiance est chargée de distribuer les attributs aux personnes concernées.

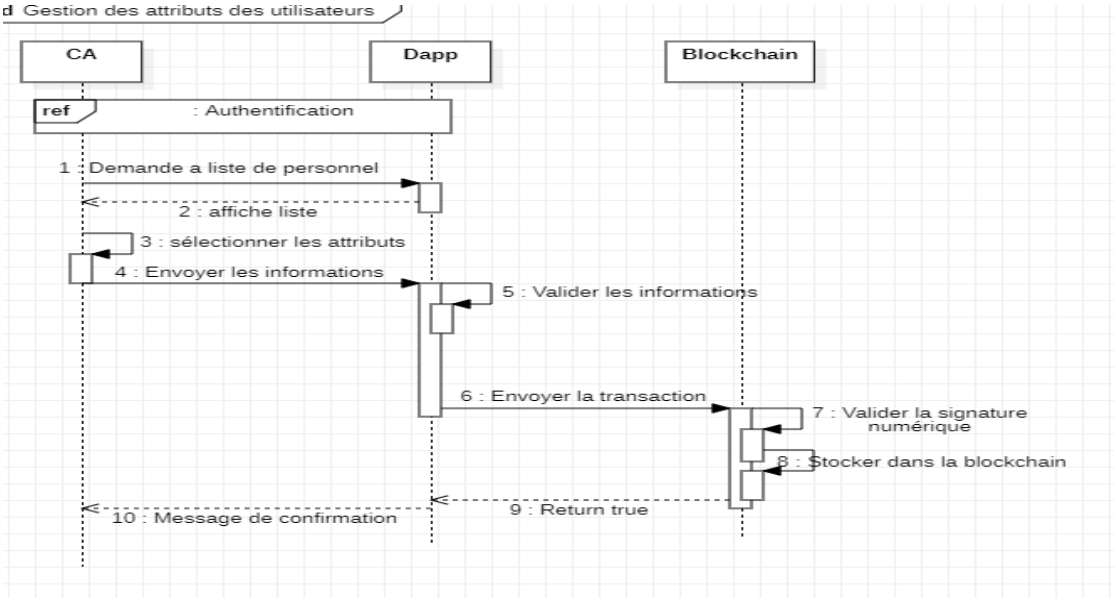

Figure 15 Diagramme de séquence pour la gestion des attributs

d. La création de politique d'accès

La politique d'accès créée par le patient est stockée dans la blockchain Ethereum. Cette politique est stockée sous forme de conditions de contrats intelligents (smart contrats) déployés sur lablockchain. Ces contrats intelligents contiennent les règles et les autorisations spécifiées par le patientpour l'accès à ses données médicales. Ainsi, lorsque les professionnels de la santé ou d'autres parties demandent l'accès aux données médicales du patient, ces contrats intelligents sont utilisés pour vérifier si l'accès est autorisé ou non en fonction des règles et des autorisations spécifiées par le patient.

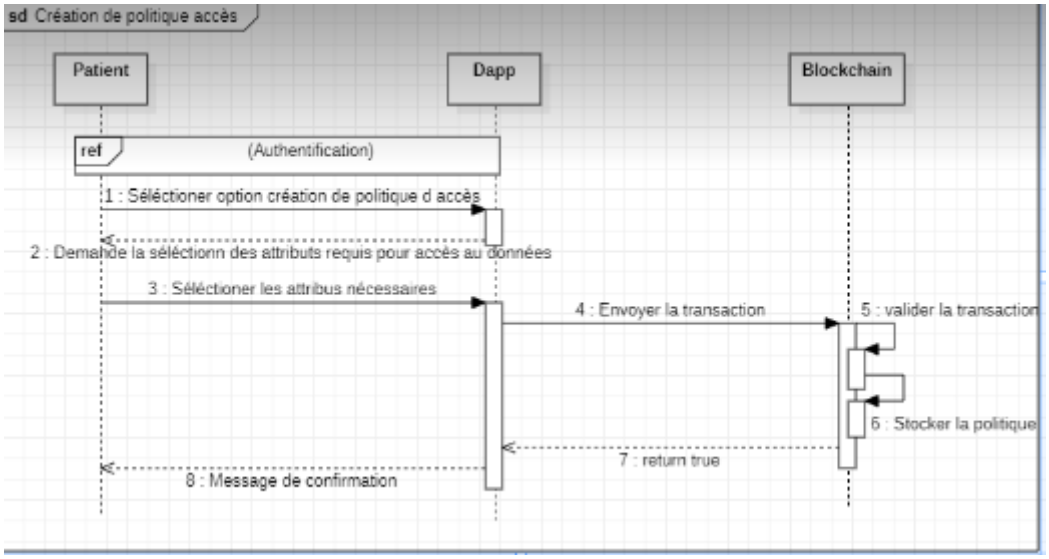

Figure 16 Diagramme de séquence pour la création de la politique d'accès pour son dossier.

e. Le chiffrement et le stockage dans la blockchain

Le stockage d'un fichier dans la blockchain implique trois étapes : la conversion du contenu du fichier en un tableau d'octets, le chiffrement de ce tableau d'octets, et enfin, la représentation du fichier chiffré sous forme d'un autre tableau d'octets qui sera stocké dans la blockchain.

Le chiffrement AES est une technique de chiffrement qui permet de chiffrer les données avec une clé secrète.

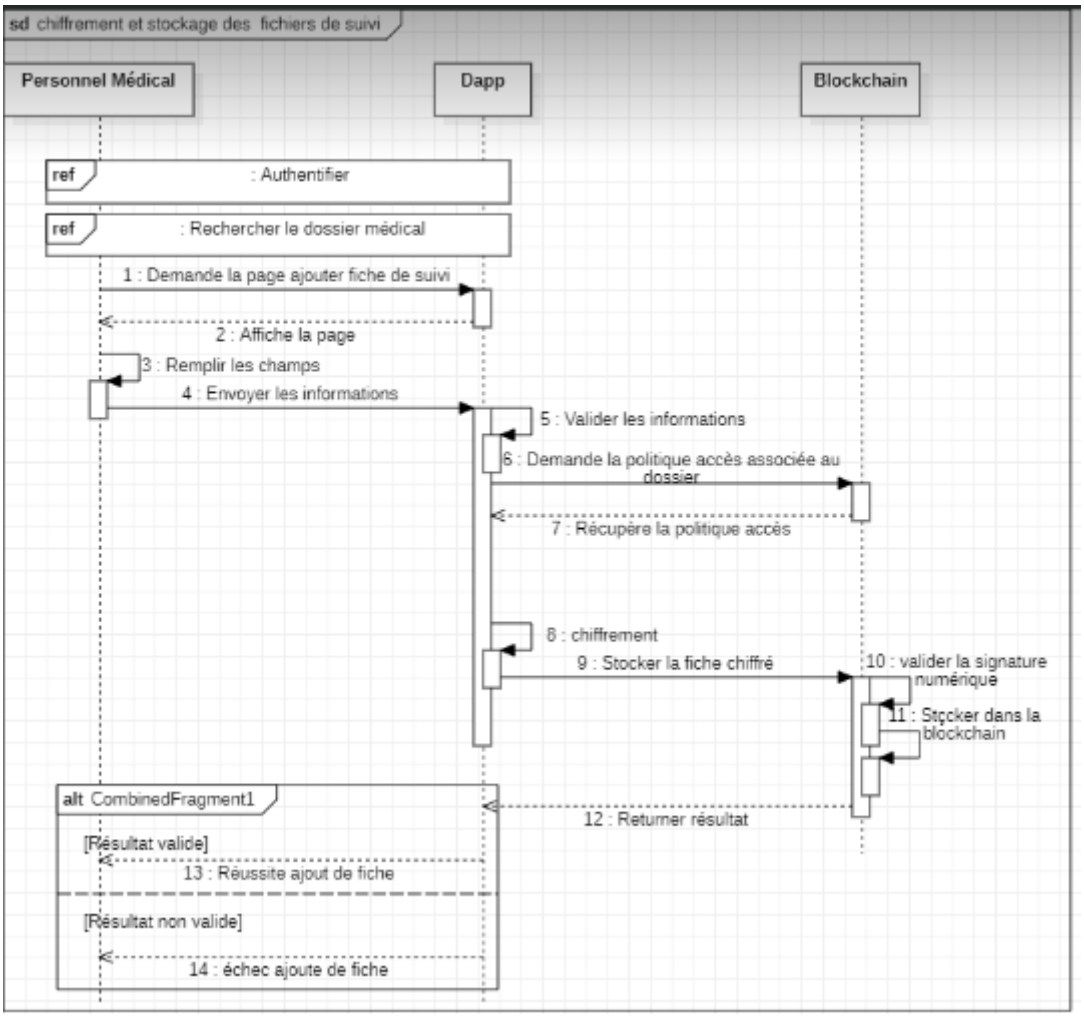

Figure 17 Diagramme de séquence de chiffrement et stockage d'une fiche de suivi

f. Téléchargement et le Déchiffrement d'un dossier médical

Seules les personnes qui possèdent les attributs nécessaires peuvent déchiffrer les données. Cette méthode est utilisée dans le but de garantir leur confidentialité et sécurité des dossiers médicaux. Lesétapes nécessaires pour réaliser ce dernier sont décrites dans le diagramme suivant :

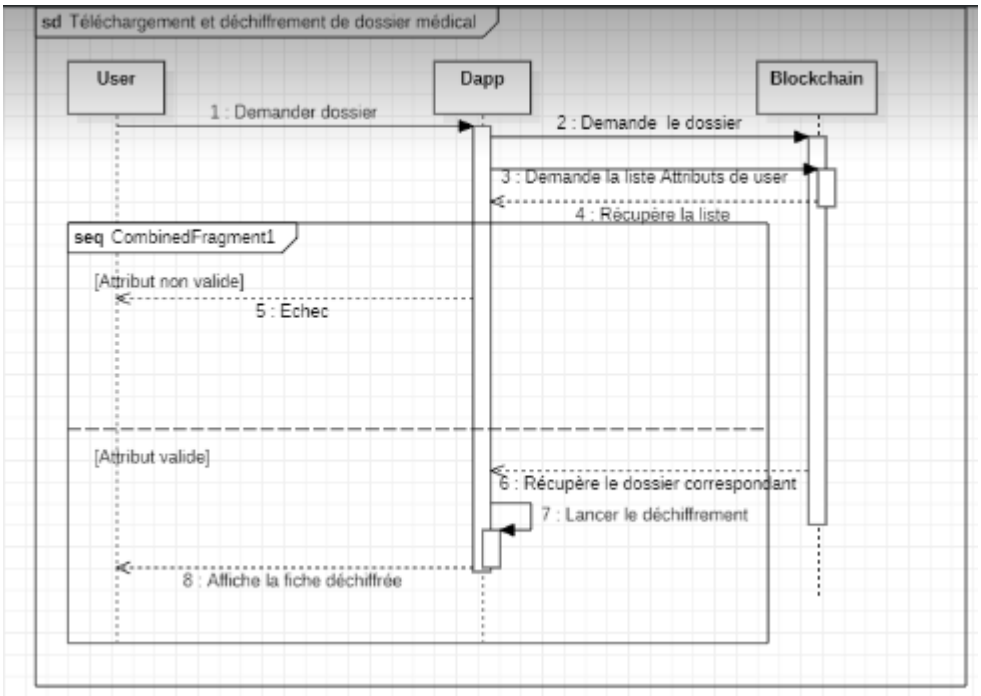

Figure 18 Diagramme de séquence pour le téléchargement et le déchiffrement du dossier médical

g. Diagramme de séquence pour l'authentification d'admin (autoritéde confiance)

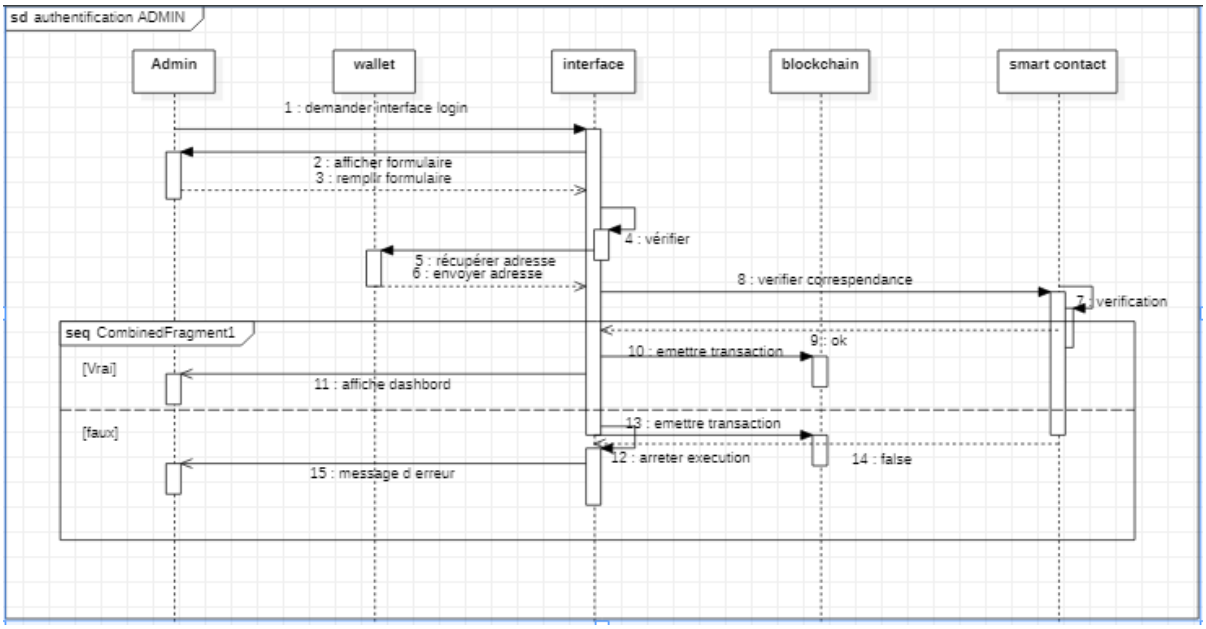

Figure 19 Diagramme de séquence authentification admin

Le diagramme de séquence ci-dessous décrit le processus

d'authentification del'administrateur, dont l'adresse Ethereum doit être prédéfinie dans le smart contrat.

Le processus commence lorsque l'administrateur entre son adresse email et son mot de passe dans l'interface utilisateur. Ensuite, l'interface utilisateur envoie ces informations dont son adresse Ethereum qui récupère de l'api au smart contrat pour vérification.

Le smart contrat vérifie alors les renseignements de l'administrateur et compare le mot de passe entré avec celui stocké dans le contrat. Si les informations sont correctes, le smart contrat renvoie une confirmation d'authentification à l'interface utilisateur, qui permet à l'administrateur d'accéder aux fonctionnalités du système réservées aux administrateurs.

En revanche, si les informations fournies ne sont pas correctes, le smart contrat renvoie un message d'erreur à l'interface utilisateur, indiquant que l'authentification a échoué.

8. Conclusion

En conclusion de la partie conception, nous avons présenté les différentes fonctionnalités et les choix de conception de notre application de gestion de données médicales basée sur la technologieblockchain et le contrôle d'accès à base d'attribut combiné avec le chiffrement AES. Nous avons également présenté les diagrammes de séquence et de cas d'utilisation pour mieux comprendre le fonctionnement de notre solution. Dans la prochaine section, nous aborderons l'implémentation de notre application en détail.

ChapitreIII :

IMPLEMENTATION

1. Introduction

Ce chapitre se concentre sur la création de l'application et décrit l'implémentation du système, les outils de développement utilisés, ainsi que quelques captures d'écran illustrant l'application réalisée.

2. Outils et Langages de programmation

Nous avons utilisé les langages et les outils suivants pour développer notre système :

2.1. Remix IDE

Nous avons utilisé une application web appelée Remix pour écrire, déboguer et déployer des contrats intelligents Ethereum. Remix est une plateforme en ligne accessible à l'adresse https://remix.ethereum.org . Grâce à cet outil, nous avons pu écrire du code en Solidity, un langage de programmation spécifique à Ethereum, puis le déployer sur la blockchain. Remix offre une interface conviviale pour faciliter le développement et la gestion des contrats intelligents.

2.2. Visual Studio Code

Nous avons utilisé un éditeur de code source multiplateforme, open source et gratuit pour développer notre système. Cet éditeur de code peut être utilisé avec différents langages de programmation tels que Java, JavaScript, Node.js et C++. Il offre une interface conviviale pour la rédaction, la modification et le débogage du code source. 2.1.

2.3. Truffle

Nous avons utilisé un pipeline d'actifs pour les blockchains basées sur la machine virtuelle Ethereum. Ce pipeline permet aux développeurs de démarrer un projet de contrat intelligent en un clic. Il fournit une structure de projet préconfigurée avec des fichiers et des répertoires qui facilitent ledéploiement et les tests. Grâce à ce pipeline, nous avons pu rapidement mettre en place notre projetde contrat intelligent et bénéficier d'une organisation efficace des fichiers et des ressources nécessaires au déploiement et à la réalisation de tests.

Pour installer truffle sous Windows :

```
C:\Users\elitebook>npm install truffle
```

2.4. Ganache

Ganache est une blockchain personnelle qui simplifie le développement d'applications distribuées basées sur Ethereum et Corda. Il permet aux développeurs de créer, déployer et tester leurs applications dans un environnement sécurisé et prévisible. Ganache UI est une application de bureau qui prend en charge les technologies Ethereum et Corda. Elle offre une interface conviviale pour configurer et gérer une blockchain personnalisée, tester des contrats intelligents et observer l'état de la blockchain en temps réel. En utilisant Ganache UI, les développeurs peuvent accélérer leur cycle de développement en disposant d'un environnement sûr et déterministe pour leurs applications distribuées. Ensuite, il est nécessaire de configurer notre blockchain afin de permettre le stockage des dossiers en augmentant la valeur du gas. L'augmentation de la valeur du gas dans Ganache peut être requise pour exécuter des transactions ou des opérations complexes qui nécessitent davantage de ressources de calcul et de stockage dans le réseau Ethereum. Cela peut être réalisé en utilisant la commande suivante :

```
C:\Users\elitebook\Desktop\MEDOC>ganache-cli --gasLimit 20000000
```

2.5. Node.js

Node.js est une plate-forme basée sur le moteur d'exécution JavaScript de Chrome, qui permet de créer facilement des applications réseau rapides et évolutives. Pour développer des contrats intelligents, nous devons configurer notre environnement. La première dépendance dont nous aurons besoin est Node Package Manager (NPM), qui est fourni avec Node.js. NPM permet de gérer les packages et les dépendances nécessaires à notre projet de développement de contrats intelligent.

La première dépendance essentielle dont nous aurons besoin est Node Package Manager (NPM), qui est inclus avec Node.js. NPM est le gestionnaire de packages pour la plateforme Node.js. Il permet d'installer et de gérer les modules nécessaires à nos projets. NPM est responsable de la résolution des dépendances, en veillant à ce que les modules requis soient installés et disponibles pour Node.js, tout en gérant les éventuels conflits de dépendances entre les modules. En résumé, NPM simplifie le processus d'ajout et de gestion

des modules dans nos projets Node.js.

```
C:\Users\elitebook>node --version
v18.16.0

C:\Users\elitebook>npm --version
9.5.1

C:\Users\elitebook>truffle -v
Truffle v5.9.0 (core: 5.9.0)
Ganache v7.8.0
Solidity - 0.8.19 (solc-js)
Node v18.16.0
Web3.js v1.10.0
```

2.6. MetaMask

MetaMask est une extension de navigateur qui permet d'accéder aux applications distribuées (DApps) basées sur Ethereum. Elle agit comme un portefeuille numérique et offre une interface conviviale pour interagir avec les DApps directement depuis votre navigateur. Grâce à MetaMask, vous pouvez effectuer des transactions, interagir avec des contrats intelligents, et gérer vos actifs numériques en toute sécurité. En résumé, MetaMask facilite l'accès et l'utilisation des applications distribuées Ethereum dans votre navigateur.

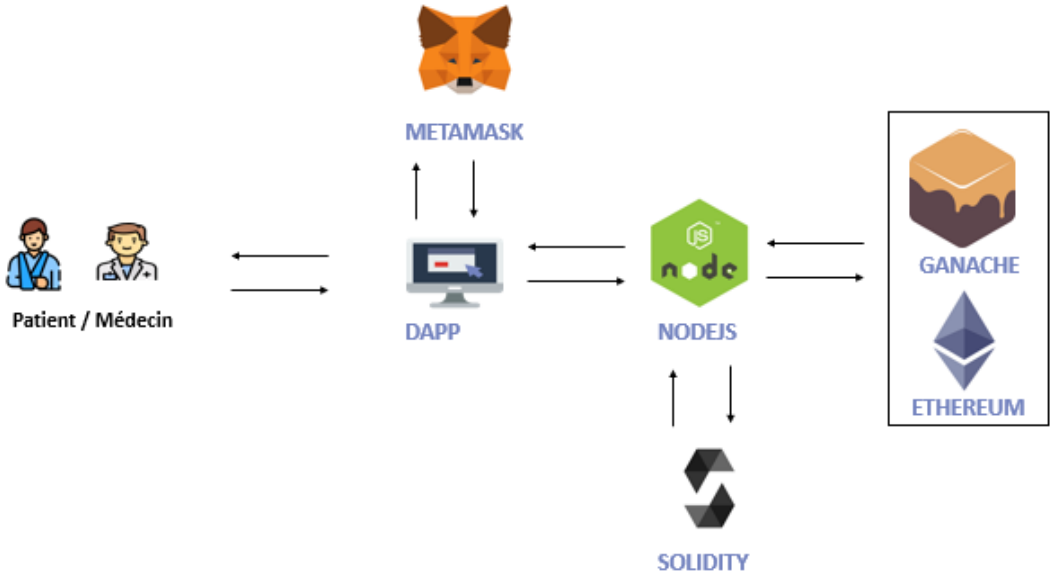

Figure 20 interaction entre les outils

3. Description du système

Notre objectif dans ce système est de concevoir une application basée sur la technologie blockchain qui permet le contrôle, le stockage et le partage facile des dossiers

médicaux électroniques des patients entre les professionnels de santé. L'application vise à faciliter l'accès aux informations médicales telles que les antécédents médicaux, les résultats d'analyses de laboratoire, l'imagerie médicale, les traitements en cours, etc. Elle offre ainsi une solution sécurisée et efficace pour le partage et la gestion des données médicales, améliorant ainsi la coordination des soins et la qualité des services de santé.

4. Configuration de l'environnement

4.1. Créer le projet MeDoc

Pour commencer, nous devons créer un dossier appelé "MeDoc" et initialiser un projet npm à l'intérieur. Pour ce faire, faut suivre les étapes suivantes dans votre terminal :

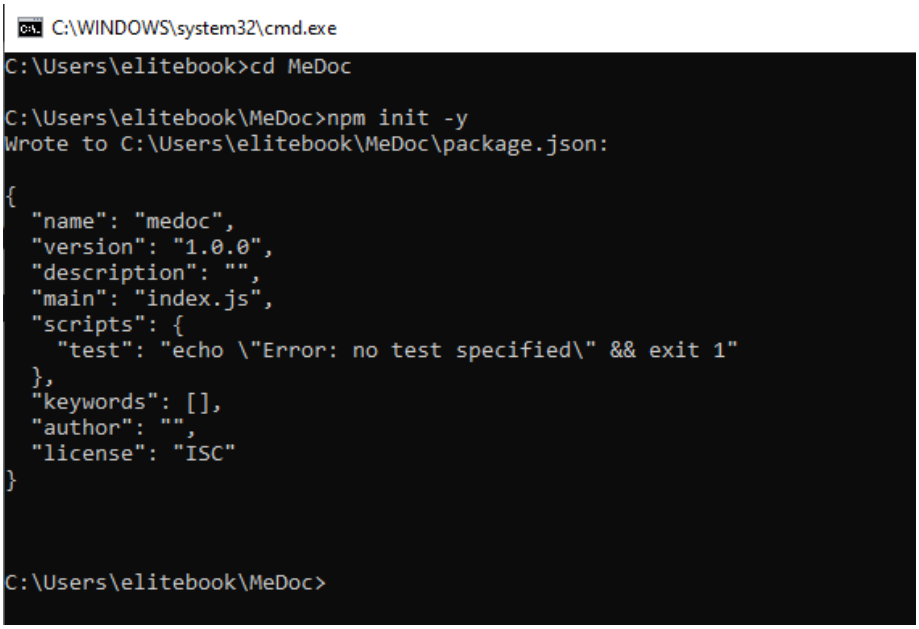

Figure 21 Création de mon projet MeDoc

La commande npm install permet d'installer un package spécifique ainsi que toutes les dépendances dont il a besoin.

```
C:\Users\elitebook\MeDoc>npm install
```

Express est un framework d'applications Web minimaliste pour Node.js. Il simplifie le processus de développement en fournissant des fonctionnalités essentielles pour la création d'applications Web, telles que la gestion des routes et des requêtes http.

```
C:\Users\elitebook\MeDoc>npm install express --save
```

Web3.js est une bibliothèque JavaScript qui facilite l'interaction avec la blockchain dans le développement de sites Web ou de clients. Elle permet d'écrire du code pour lire et écrire des données à partir de la blockchain à l'aide de contrats intelligents.

```
C:\Users\elitebook\MeDoc>npm install web3 --save
```

- La fonction loadWeb3() configure Web3 pour permettre la communication avec la blockchain.
- La ligne de code contract = new web3.eth.Contract(contractAbi, contractAddress) crée une instance de contrat intelligent avec Web3.js. Cette instance est utilisée pour lire les données des contrats intelligents en utilisant Web3.js.

4.2. Vérification de package.json

Après les installations, nous ouvrirons le dossier du projet nommé "MeDoc" dans VSCode et vérifierons le contenu du fichier package.json. Ce fichier est généré automatiquement par npm lors de l'initialisation du projet et il contient différentes configurations, y compris la liste des packages que nous avons installés précédemment.

Figure 22 package.json

4.3. Développer notre projet

Lorsque nous exécutons la commande truffle init, cela configure la structure de base dans notre répertoire de projet.

```
PS C:\Users\elitebook\MeDoc> truffle init
```

On va voir :

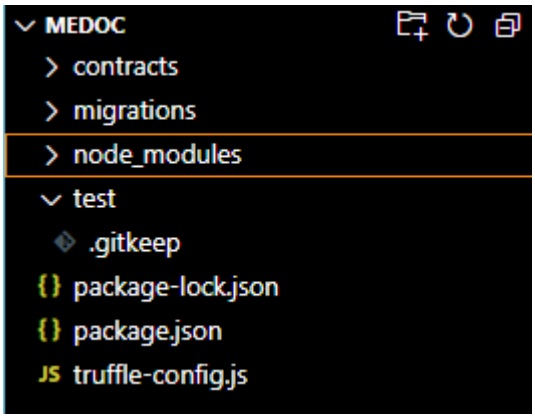

Figure 23 Structure du répertoire

4.4. Création des contract intelligents

Pour créer une application de gestion de dossiers médicaux décentralisée, nous commençons par créer les contrats intelligents Ethereum dans le répertoire "Contracts". À l'intérieur de chaque fichier de contrat, nous définissons les variables et les fonctionnalités nécessaires, et nous fournissons le code pour la structure de chaque fonction utilisée dans notre application :

- Le smart contract médecin qui contient son nom, prénom, email, mot de passe, Numéro d'Identification du Personnel (NIP), sa profession, son service, son adresse Ethereum, ainsi que des indicateurs de statut pour l'enregistrement et la validation.

Le mapping "doctors" est utilisé pour associer chaque adresse Ethereum à un objet de type "Doctor", permettant ainsi de stocker les informations des médecins dans la blockchain.

```
contract Doctors {
    struct Doctor {
        string firstName;
        string lastName;
        string email;
        string password;
        string NIP;
        string profession;
        string service;
        address doctorAddress;
        bool isRegistered;
        bool isValidated;
        string[] attributs;
    }

    mapping(address => Doctor) public doctors;
    address[] public doctorAddresses;
    address[] public registeredDoctors;
```

Figure 24 structure et mapping du smart contract Médecin

- Le smart contract des patients qui contient son identificateur, nom, prénom, email, mot de passe, date de naissance, son adresse, ainsi que des indicateurs de statut pour l'enregistrement et la validation.

Le mapping "patients" qui utilise les adresses des patients comme clé pour stocker toutes les données des patients.

```
contract Patients {
    struct Patient {
        uint256 id;
        string firstName;
        string lastName;
        string email;
        string password;
        string birthday;
        address patientAddress;
        bool isValidated;
        bool isRegistered;
    }

    mapping(address => Patient) private patients;
    address[] public patientAddresses;
    uint256 public patientCount;
```

Figure 2535structure et mapping du smart contract Patient

- Le smart contract du dossier médical : représente un dossier médical individuel et contient plusieurs champs tels que patientId, bloodGroup, medicalHistory et AccessPolicy. Il utilise également deux mappings : "medicalTests" qui associe des noms de tests médicaux à leurs données (sous forme de tableau d'octets), et

"followUpSheets" qui associe des identifiants de suivi à leurs feuilles de suivi (également sous forme de tableau d'octets).

```
contract MedicalRecords {
    struct MedicalRecord {
        uint256 patientId;
        string bloodGroup;
        string medicalHistory;
        string accessPolicy;
        mapping(string => bytes) medicalTests;
        mapping(uint256 => bytes) followUpSheets;
    }

    mapping(uint256 => MedicalRecord) private medicalRecords;
    uint256 public patientCount;
```

Figure 26 structure et mapping du smart contract administrateur

Une fois crée, les contrats doivent être compilés et sauvegardés pour une utilisation ultérieure. Commela montre la figure suivante :

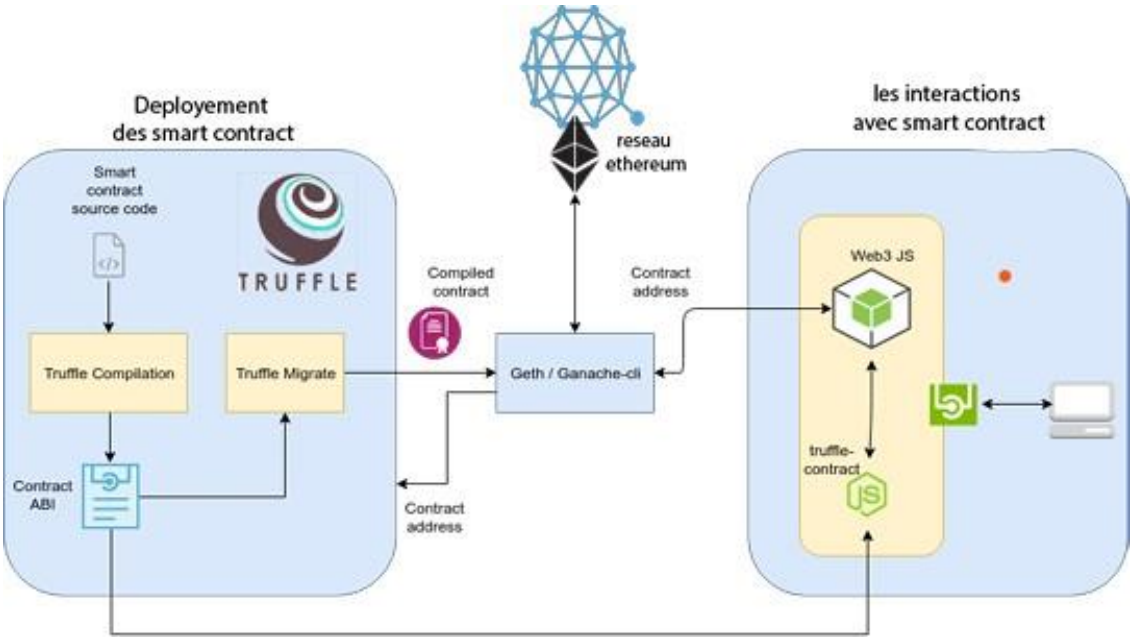

Figure 27 deployement des contract intelligents

Dans un premier temps, nous avons envisagé d'utiliser le chiffrement CP-ABE pour sécuriser les dossiers médicaux. Cependant, en raison de contraintes de ressources et de l'indisponibilité d'une bibliothèque CP-ABE compatible avec Windows, nous avons opté pour une alternative, à savoir le chiffrement Advanced Encryption Standard AES (128 bits). Afin de

mettre en œuvre cette alternative, il est nécessaire d'installer la bibliothèque crypto-js en utilisant la commande suivante :

```
C:\Users\elitebook>npm install crypto-js
```

5. Présentation de la plateforme

5.1. Page d'inscription :

Sur cette page de notre application, les utilisateurs, qu'ils soient médecins ou patients, ont la possibilité de s'inscrire en remplissant leurs informations respectives sur leurs pages d'inscription dédiées. Une fois que les utilisateurs ont soumis leurs informations, celles-ci sont examinées et validées par l'administrateur du système.

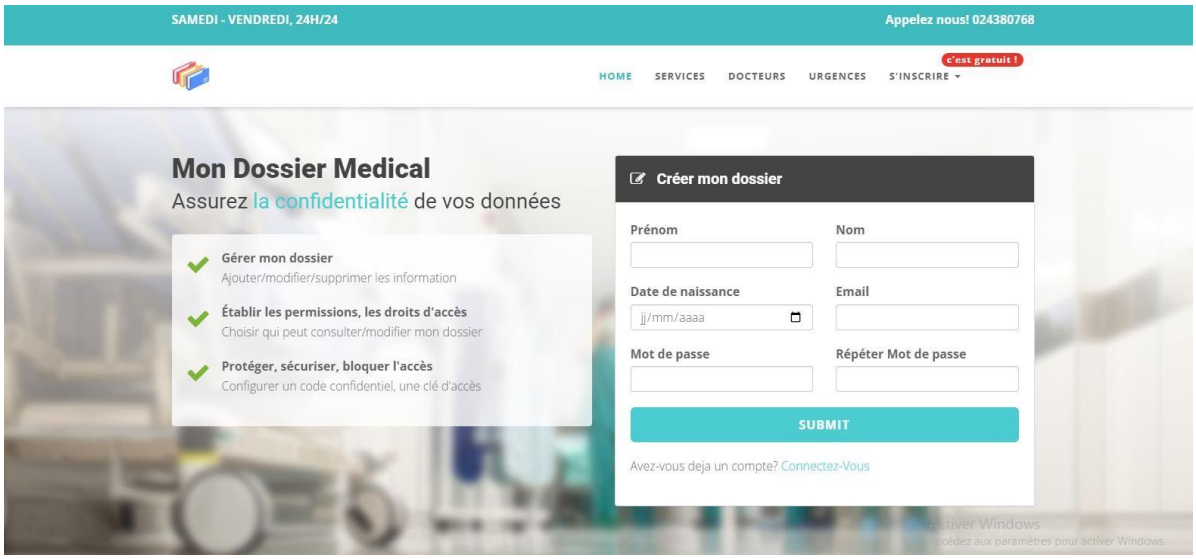

Figure 28 Page d'inscription patient

5.2. Page d'authentification

Après avoir été approuvée par l'administrateur, la demande d'inscription de l'utilisateur lui permettra de s'authentifier en utilisant son adresse e-mail et son mot de passe. L'utilisateur devra également préciser s'il s'agit d'un médecin ou d'un patient lors de la procédure d'authentification.

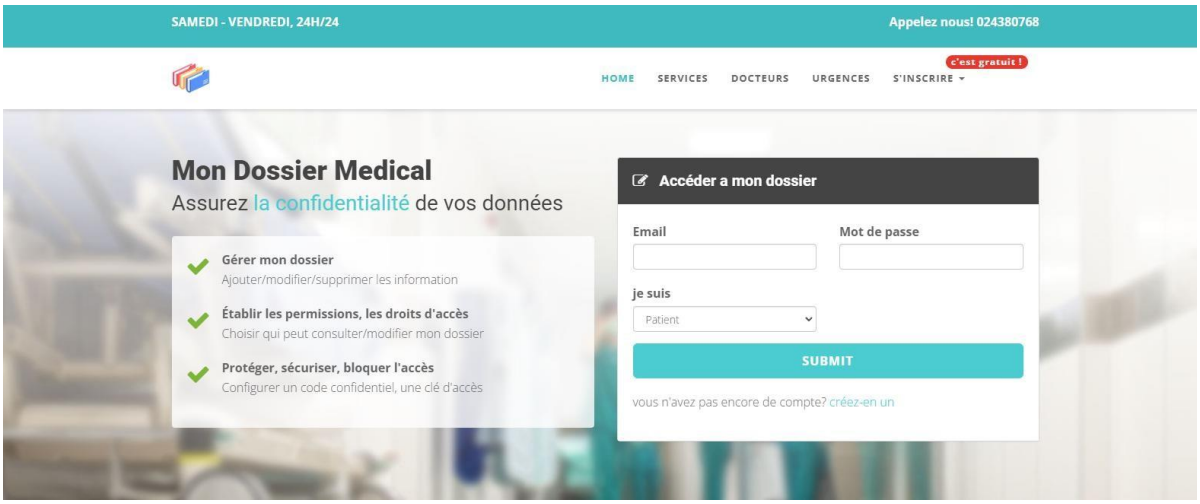

Figure 29 page d'authentification

5.3. Profil

Dans le profil du patient, il est possible d'enregistrer des informations telles que le groupe sanguin, les antécédents médicaux ainsi que les attributs des médecins autorisés à consulter son dossier médical. Cette fonctionnalité permet au patient de fournir des détails importants sur sa santé, facilitant ainsi la prise en charge médicale. De plus, en spécifiant les médecins autorisés, le patient maintient le contrôle sur l'accès à ses informations confidentielles, assurant ainsi la confidentialité de son dossier médical.

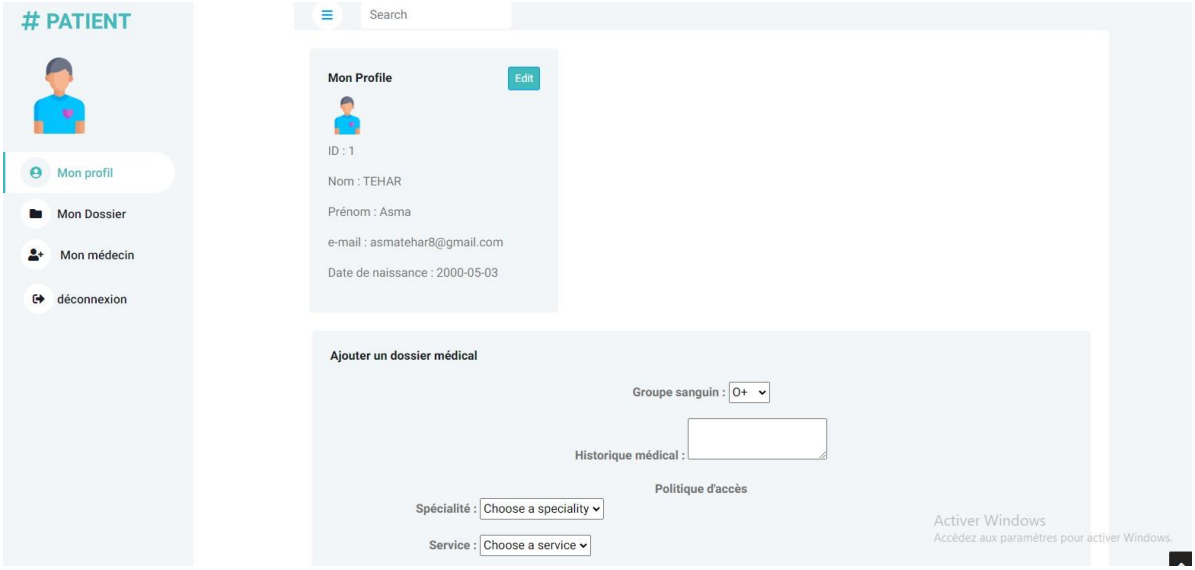

Figure 30profil du patient

5.4. Profil de l'administrateur

L'administrateur dispose de la possibilité d'ajouter un nouveau membre administrateur au systèmeen saisissant son nom complet, son adresse e-mail, son mot de passe, ainsi que son adresse du portefeuille Ethereum.

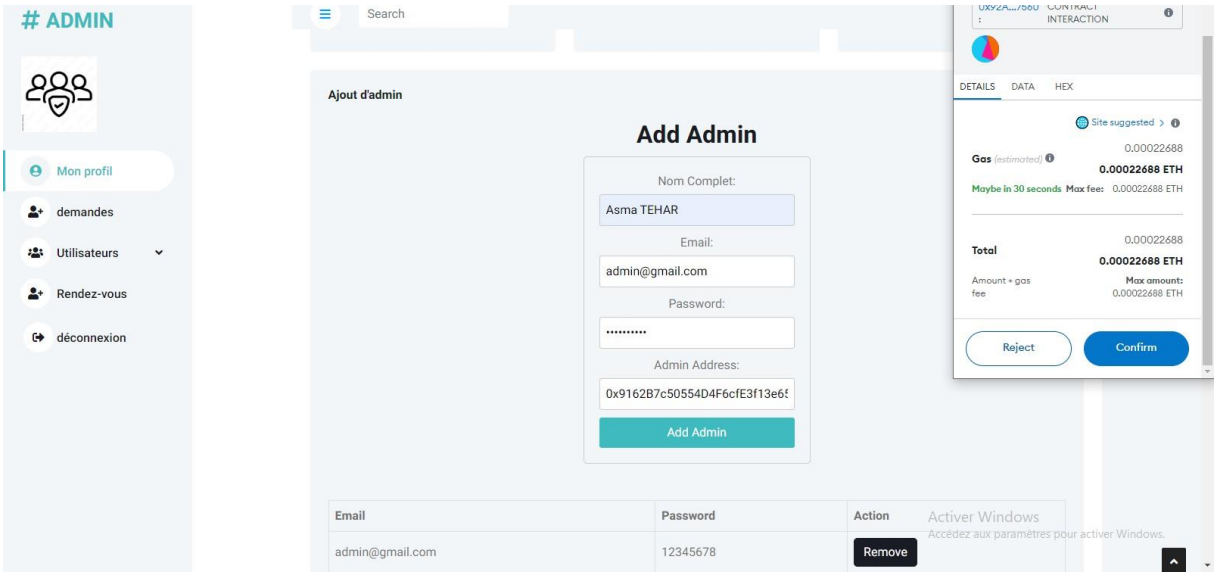

Figure 31 Ajouter un administrateur

L'administrateur dispose également de la possibilité de définir les attributs spécifiques de chaque médecin admis dans la plateforme. Cette fonctionnalité permet de personnaliser les informations associées à chaque médecin, en fonction de ses qualifications, de son domaine d'expertise ou d'autres critères pertinents.

Figure 32 Gestion d'attribut par l'autorité de confiance

5.5. Ajouter une fiche de suivie

Une fois la consultation terminée, le médecin a le droit d'ajouter une fiche de suivi au dossier du patient. Cette fiche peut contenir des remarques pertinentes et spécifiques à la

consultation. De plus, le médecin peut également ajouter d'autres documents au format PDF, tels que les résultats d'analyses, les comptes rendus de consultation ou les correspondances entre les médecins.

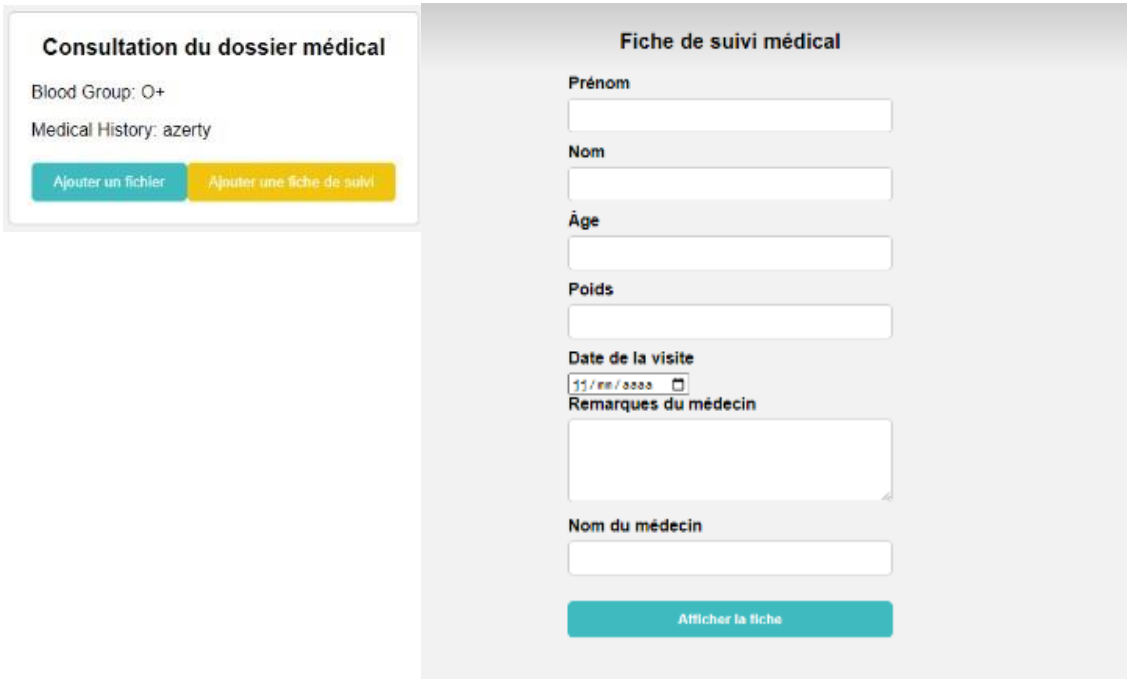

Figure 33 ajouter fiche de suivi ou un autre document

6. Conclusion

Dans ce chapitre, nous avons exposé les diverses technologies mises en œuvre dans le cadre de notre projet, ainsi que leur utilisation en conformité avec notre approche conceptuelle. Le blockchain Ethereum a été utilisé pour le stockage des informations d'authentification, les attributs employés dans le control d'accès ABAC et les dossiers médicaux. De plus, les dossiers médicaux ont été chiffrés à l'aide de l'algorithme AES. En d'autres termes, nous avons réussi à achever la première version de notre application, en adhérant au plan de conception initial, tout en optant l'algorithme AES pour le chiffrement, plutôt que le chiffrement cp-abe comme initialement envisagé.

Conclusion générale et prescriptives

L'objectif de cet article est d'apporter une solution au problème de sécurité des données médicales stockées dans les applications de e-santé en permettant leur diffusion aux utilisateurs tout en préservant la confidentialité, l'intégrité et la confidentialité des données. Pour atteindre cet objectif, nous avons mené une étude de la littérature en trois parties.

-Dans une première partie, nous avons étudié les applications blockchain pour démontrer leur utilité dans la protection des informations.

- La partie 2 couvrait les concepts de cryptage et de contrôle d'accès pour ajouter une couche de sécurité supplémentaire aux données stockées sur la blockchain. Nous nous sommes particulièrement intéressés au contrôle d'accès basé sur les attributs (ABAC) et au chiffrement basé sur les politiques d'attributs (CP-ABE).

- Enfin, la troisième partie a exploré les concepts d'hôpitaux intelligents et de dossiers médicaux électroniques, en mettant l'accent sur leurs exigences de sécurité.

Nous avons donc proposé une plateforme de e-santé combinant deux approches de sécurité.

- Une blockchain privée qui enregistre et vérifie toutes les transactions de l'application et stocke en toute sécurité les dossiers médicaux

- Contrôle d'accès ABAC utilisant l'algorithme de cryptage AES qui crypte les fichiers avant de les stocker sur la blockchain.

La combinaison de ces approches nous a permis d'atteindre les objectifs de sécurité nécessaires dans ce domaine.

Disponibilité : l'utilisation de la technologie blockchain exécutée sur plusieurs nœuds garantit une très haute disponibilité des informations par rapport aux autres technologies.

Contrôle d'accès : vous pouvez spécifier qui est autorisé à accéder à des dossiers médicaux spécifiques.

Confidentialité : Les données de la blockchain sont cryptées et ne sont accessibles qu'aux personnes autorisées.

Les avantages de la technologie blockchain et son immuabilité garantissent l'intégrité des

données.

Non-répudiation : La blockchain stocke toutes les transactions qui ne peuvent pas être modifiées, donc aucune transaction dans le système n'est rejetée.

Traçabilité : suivez le statut et le mouvement des informations.

Pour mettre en œuvre la solution, nous avons développé une application Node.js qui utilise la blockchain Ethereum et la bibliothèque de chiffrement AES. Notre application offre les fonctionnalités suivantes :

- Authentification de l'utilisateur avec nom d'utilisateur/adresse e-mail et mot de passe.
- Gestion des demandes d'inscription et des attributs des médecins par les autorités de confiance.
- Ajout du formulaire de suivi médical. Ces fichiers sont cryptés avant d'être stockés sur la blockchain.
- Définition des accès des patients au dossier médical et des droits d'accès à ce dossier.

Cependant, la plateforme peut encore être améliorée. Par exemple :

- Ajoutez plus d'acteurs à notre application. Infirmières, soignants, représentants légaux des patients et autres.
- Permet l'importation de documents tels que des radiographies, des IRM, des scanners.
- Étendre le réseau blockchain pour inclure différents hôpitaux dans différents pays tout en préservant la confidentialité des données
. - Intégrer l'Internet des objets (IoT) pour fournir des données de santé en temps réel telles que : Obtenir les signes vitaux des patients et les inclure dans les systèmes de contrôle d'accès pour une autorisation spécifique basée sur des conditions en temps réel.
- Mettre en œuvre des techniques d'apprentissage automatique pour l'analyse des données médicales et la détection des anomalies afin d'améliorer la précision du diagnostic et la prise de décision médicale.
- Découvrez des solutions de sécurité avancées telles que la biométrie pour renforcer l'authentification des utilisateurs et empêcher les accès non autorisés.

Ces améliorations supplémentaires améliorent la fonctionnalité, la sécurité et l'efficacité de notre plateforme, offrant une expérience plus riche et plus sécurisée à nos utilisateurs.

Réferences

[1] L. Leloup, Blockchain : La révolution de la confiance, Editions Eyrolles, 2017.

[2] S. Nakamoto, «Bitcoin: A Peer-to-Peer Electronic Cash System,» 2008.

[3] B. Müßigmann, H. von der Gracht et E. Hartmann, «Blockchain technology in logistics and supply chain management—A bibliometric literature review from 2016 to January 2020,» *IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT,* 2020.

[4] M. Goyal et A. Kumar, «Sustainable E-Infrastructure for Blockchain-Based Voting System,» chez *Digital Cities Roadmap: IoT-Based Architecture and Sustainable Buildings*, 2021, pp. 221-251.

[5] S. Verma et A. Sheel, «Blockchain for government organizations: Past, present and future,» *Journal of Global Operations and Strategic Sourcing,* vol. 15, n° %13, pp. 406-430, 2022.

[6] C. F. Plisson, «CLa blockchain, un bouleversement économique, juridique voire sociétal,» *I2D - Information, données & documents,,* vol. 54, pp. 20-22, 2017.

[7] A. Shahnaz, U. Qamar et A. Khalid, «Using Blockchain for Electric Health Records,» *IEEE Access,* vol. 7, pp. 147783-147795, 2019.

[8] L. Ghiro, F. Restuccia, S. D'Oro et S. Ba, «What is a Blockchain? A Definition to Clarify the Role of the Blockchain in the Internet of Things,» *2021,19th Mediterranean Communication and Computer Networking Conference (MedComNet),* pp. 1-8, 2021.

[9] J. J. Sikorski, J. Haughton et M. Kraft., «Blockchain technology in the chemical industry: Machine-to-machine electricity market,,» vol. 195, pp. 234-246, 2017.

[10] C. F. Plisson, «LA BLOCKCHAIN, UN BOULEVERSEMENT ÉCONOMIQUE, JURIDIQUE VOIRE SOCIÉTAL.,» *2D - Information, données & documents,,* vol. 54, pp. 20-22, 2017.

[11] «Bitcoin est un réseau de paiement novateur et une nouvelle forme d'argent.,» [En ligne]. Available: https://bitcoin.org/fr/. [Accès le 27 01 2023].

[12] «About Ripple RippleNet.,» Ripple Labs Inc, 2020. [En ligne]. Available: https://ripple.com/ripplenet/. [Accès le 29 01 2023].

[13] A. BENIICHE, «A study of blockchain oracles,» *arXiv preprint arXiv:2004.07140,* 2020.

[14] V. Acharya, A. Eswararao Yerrapati et N. Prakash, Oracle Blockchain Services Quick Start Guide:A practical approach to implementing blockchain in your enterprise, Packt Publishing, 2019.

[15] J. J. Sikorski, J. Haughton et M. Kraft., «Blockchain technology in the chemical industry: Machine-to-machine electricity market,» vol. 195, pp. 234-246, 2017.

[16] Z. Zheng, S. Xie, H. Dai, X. Chen et H. Wang, «An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends,» 2017 IEEE 6th International Congress on Big Data, 2017.

[17] K. Pardeshi, «REVIEW OF BLOCKCHAIN ARCHITECTURE A SURVEY,» *Composition Theory,* vol. 13, n° %14, pp. 239-248, 2021.

[18] R. Agrawal, J. Chatterjee, A. Kumar et P. Singh Rathore, Blockchain Technology and the Internet of Things: Challenges and Applications in Bitcoin and Security., Apple Academic Press, 2021.

[19] C. F. Plisson, «LA BLOCKCHAIN, UN BOULEVERSEMENT ÉCONOMIQUE, JURIDIQUE VOIRE SOCIÉTAL.,» 2D - Information, données & documents,, vol. 54, pp. 20-22, 2017

[20] C.Vijai, M. Elayaraja, S.M.Suriyalakshmi et D.Joyce, «The Blockchain Technology and Modern Ledgers Through Blockchain Accounting,» SSRN Electronic Journal, vol. 18, 2019

[21] B. Becher, «What Are Blockchain Nodes and How Do They Work?,» Built In, 29 Septembre 2022. [En ligne]. Available: https://builtin.com/blockchain/blockchain-node. [Accès le 17 01 2023].

[22] S. Dramé-Maigné, « Blockchain and access control: Towards a more secure Internet of Things,» chez École doctorale n°580 : Sciences et Technologies de l'Information et de la Communication (STIC), Paris, 2019.

[23] L. Lars, « Qu'est-ce qu'un bloc dans la technologie blockchain ?» CRYPTOAST, 27 07 2020. [En ligne]. Available: https://cryptoast.fr/bloc-blockchain-crypto-explication/. [Accès le 27 02 2023].

[24] J. Maldonado, «Qu'est-ce que l'horodatage?,» Bit2me Academy, 12 12 2019. [En ligne]. Available: https://academy.bit2me.com/fr/timestamp-blockchain/. [Accès le 27 02 2023].

[25] N. Szabo, «Smart contracts: building blocks for digital markets,» EXTROPY: The Journal of Transhumanist Thought,(16), vol. 18, n° %12, p. 28, 1996.

[26] D. Macrinici, C. Cartofeanu et S. Gao, «Smart contract applications within blockchain technology: A systematic mapping study,» Telematics and Informatics, vol. 35, n° %18, pp. 2337-2354, 2018.

[27] R. Jafri et S. Singh, «Blockchain applications for the healthcare sector: Uses beyond Bitcoin,» chez Blockchain Applications for Healthcare Informatics., Uttar Pradesh, India, Elsevier Inc., 2022, pp. 71-91

[28] G. Lowe, «Casper : A Compiler for the Analysis of Security Protocols,» chez *Proceedings 10th Computer Security Foundations Workshop*, 1997, pp. 18-30.

[29] M. PEASE, R. SHOSTAK et L. LAMPORT, «Reaching agreement in the presence of faults,» *Journal of the AssoctaUon for Compmmg Machinery,* vol. 27, n° %12, pp. 228-234, 1980.

[30] S. King, «Primecoin: Cryptocurrency with prime number proof-of work,» 2013

[31] A. Barhanpure, P. Belandor et B. Das, «Proof of stack consensus for blockchain networks,» chez Communications in Computer and Information Science, 2019, p. 104–116.

[32] G. Lowe, «Casper : A Compiler for the Analysis of Security Protocols,» chez Proceedings 10th Computer Security Foundations Workshop, 1997, pp. 18-30.

[33] M. PEASE, R. SHOSTAK et L. LAMPORT, «Reaching agreement in the presence of faults,» Journal of the AssoctaUon for Compmmg Machinery, vol. 27, n° %12, pp. 228-234, 1980.

[34] R. A. Michelin, A. Dorri, M. Steger, R. C. Lunardi, S. S. Kanhere, R. Jurdak et A. F. Zorzo, «SpeedyChain: A framework for decoupling data from blockchain for smart cities,» chez Proceedings of the 15th EAI international conference on mobile and ubiquitous systems: Computing, networking and services, 2018, pp. 145-154.

[35] J. Huang, L. Kong, G. Chen, M.-Y. Wu, X. Liu et P. Zeng, «Towards secure industrial IoT: Blockchain system with credit-based consensus mechanism,» IEEE Transactions on Industrial Informatics, vol. 15, n° %16, pp. 3680-3689, 2019.

[36] T. Alladi, V. Chamola, R. M. Parizi et K.-K. R. Choo, «Blockchain applications for industry 4.0 and industrial IoT: A review,» IEEE Access, vol. 7, pp. 176935-176951, 2019.

[37] M. Waseem, M. Adnan Khan, A. Goudarzi, S. Fahad, I. A. Sajjad et P. Siano, «Incorporation of Blockchain Technology for Different Smart Grid Applications: Architecture, Prospects, and Challenges,» Energies, vol. 16, n° %12, p. 820, 2023.

[38] S. Biswas, K. Sharif, F. Li, S. Maharjan, S. P. Mohanty et Y. Wang, «PoBT: A lightweight consensus algorithm for scalable IoT business blockchain,» IEEE Internet of Things Journal, vol. 7, n° %13, pp. 2343-2355, 2019.

[39] M. Waseem, Z. Lin, S. Liu, Z. Jinai, M. Rizwan et I. A. Sajjad, «Optimal BRA based electric demand prediction strategy considering instance-based learning of the forecast factors,» International Transactions on Electrical Energy Systems, vol. 31, n° %19, p. e12967, 2021.

[40] M. Narouei, H. Takabi et R. Nielsen, «Automatic Extraction of Access Control Policies from Natural Language Documents,» IEEE Transactions on Dependable and Secure Computing, vol. 17, n° %13, pp. 506-517, 2020

[41] P. Biswas, R. Sandhu et R. Krishnan, «Label-based access control: An ABAC model with enumerated authorization policy,» chez Proceedings of the 2016 ACM International Workshop on Attribute Based Access Control, 2016, pp. 1-12.

[42] B. Lutkevich, «DEFINITION access control,» TechTarget, [En ligne]. Available: https://www.techtarget.com/searchsecurity/definition/access-control. [Accès le 24 02 2023].

[43] K. Kane et J. C. Browne, «On classifying access control implementations for distributed systems,» chez Proceedings of the eleventh ACM symposium on Access control models and technologies, 2006, pp. 29-38.

[44] L. Wang, D. Wijesekera et S. Jajodia, «A logic-based framework for attribute based access control,» chez Proceedings of the 2004 ACM workshop on Formal methods in security engineering, 2004, pp. 45-55

[45] R. Masood et M. A. a. o. Shibli, «Comparative analysis of access control systems on cloud,» chez 2012 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, IEEE, 2012, pp. 41-46.

[46] E. Yuan et J. Tong, «Attributed based access control (ABAC) for web services,» chez IEEE International Conference on Web Services (ICWS'05), IEEE, 2005.

[47] A. Sahai et B. Waters, «Fuzzy identity-based encryption,» chez Advances in Cryptology--EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005. Proceedings 24, Springer, 2005, pp. 457-473.

[48] M. Pirretti, P. Traynor, P. McDaniel et B. Waters, «Secure attribute-based systems,» chez Proceedings of the 13th ACM conference on Computer and communications security, 2006, pp. 99-112.

[49] V. Goyal, O. Pandey, A. Sahai et B. Waters, «Attribute-based encryption for fine-grained access control of encrypted data,» chez Proceedings of the 13th ACM conference on Computer and communications security, 2006, pp. 89-98.

[50] Z. Qiao, S. Liang, S. Davis et H. Jiang, «Survey of attribute based encryption,» chez 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), IEEE, 2014, pp. 1-6.

[51] B. Pradhan, B. Singh, A. Bhoria et A. K. Singh, «A comparative study on cipher text policy attribute based encryption schemes,» International Journal of Engineering Research \& Technology, 2021

[52] Kumar, S., Bharti, A. K., & Amin, R. (2021). Decentralized secure storage of medical records using Blockchain and IPFS: A comparative analysis with future directions. Security and Privacy, 4(5), e162

[53] Aouali, J. (2021, December 12). InterPlanetary File System (IPFS), le réseau de partage de fichiers distribué qui fonde les bases du Web 3.0. Retrieved June 25, 2023, from https://cryptoast.fr/interplanetary-file-system-ipfs-reseau-partage-fichiers-web-3/#obstacles-ipfs

[54] « ISO/DTR 20514, Health Informatics – Electronic Health Record – Definition, Scope, and Context,» 2004.

[55] J. L. Sànchez, S. Savin et V. Vasileva, «Key success factors in implementing electronic medical records in University Hospital of Rennes,» L'Ecole Nationale de la Santé Publique (National School of Public Health), Rennes, Rennes, France, pp. 1-59, 2005

[56] J. J.Hathaliya et T. Sudeep, «An exhaustive survey on security and privacy issues in Healthcare 4.0,» Computer Communications, vol. 153, pp. 311-335, 2020

[57] T. Gea, . J. Paradells , M. Lamarca et Rold, «Smart Cities as an Application of Internet of Things: Experiences and Lessons Learnt in Barcelona,» chez 2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2013, pp. 552-557.

[58] J. Daemen, V. Rijmen, The Design of Rijndael, AES: The Advanced Encryption Standard, ISBN 3540425802, 2001. URL https://autonome-antifa.org/IMG/pdf/Rijndael.pdf

[59] AZZOUZI, O. Système embarqué flexible pour un chiffrement hybride symétrique/asymétrique (Doctoral dissertation, Ecole Nationale Supérieure d'informatique).

[60] Paar, C., & Pelzl, J. (2009). Understanding cryptography: a textbook for students and practitioners. Springer Science & Business Media.

[61] L'AES : Advanced Encryption Standard. 04 Novembre 2001 . [En ligne]. Available : https://www.securiteinfo.com/cryptographie/aes.shtml [Accès le 24 02 2023].