

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR,
DE LA RECHERCHE ET DE L'INNOVATION

SECRÉTARIAT GÉNÉRAL

UNIVERSITÉ JOSEPH KI-ZERBO (UJKZ)

INSTITUT BURKINABÈ DES ARTS ET MÉTIERS
(IBAM)



BURKINA FASO

La Patrie ou la Mort, nous Vaincrons

MÉMOIRE POUR L'OBTENTION DU MASTER EN INFORMATIQUE
OPTION : INGENIERIE DES SYSTEMES D'INFORMATION EN ENTREPRISE

THÈME :

**Authentification de documents administratifs
à l'aide de la blockchain**

Réalisé par : HIEN Zilèdem Pierre Canisius

Soutenu publiquement le : .../06/2025

Jury de soutenance :

Président :

Directeur de mémoire : M. Yaya TRAORE, *MC en informatique.*

Evaluateur :

Année académique : 2023-2024

DÉDICACE

A ma fille, W. Candice Urielle !

REMERCIEMENTS

Nous tenons tout d’abord à rendre grâce à Dieu Le Tout Puissant, par qui nous vivons.

Au cours de ce cycle de Master, nous avons eu le privilège de collaborer avec des personnes de marques qui nous ont fourni tout le nécessaire afin que nous parvenions à ce résultat. C’est donc le lieu pour nous, à travers ces lignes, de leur traduire notre profonde gratitude.

Nos distincts remerciements vont particulièrement à l’endroit de :

- **Dr. Yaya TRAORE**, notre Directeur de mémoire, qui a bien voulu nous encadrer sans hésitation, et pour sa disponibilité malgré un calendrier chargé. Ses critiques constructives et ses partages d’expériences nous ont été d’une très grande utilité.
- **Pr. Sadouanouan MALO**, pour ses conseils bien avisés, ses multiples encouragements et accompagnements.
- **Monsieur le Directeur de l’IBAM, le corps professoral et tout le personnel de l’IBAM** pour la formation reçue et leur accompagnement.
- **notre famille, nos collègues et camarades** pour leurs soutiens et encouragements.
- **toutes les personnes dont les noms n'ont pu être cités.**

RÉSUMÉ

ABSTRACT

TABLES DES MATIÈRES

DÉDICACE	i
REMERCIEMENTS	ii
RÉSUMÉ	iii
ABSTRACT	iv
TABLES DES MATIÈRES	v
LISTE DES FIGURES	vii
LISTE DES TABLEAUX	viii
LISTE DES SIGLES ET ABREVIATIONS	ix
CHAPITRE 1 : INTRODUCTION GENERALE	2
1.1. Contexte et justification	2
1.2. Problématique et hypothèses	4
1.3. Objectif du sujet	5
1.4. Résultats attendus	5
1.5. Organisation du travail	6
CHAPITRE 2 : TECHNOLOGIE BLOCKCHAIN	9
2.1 Historique et définitions de la blockchain	9
2.1.1. Historique de la blockchain	9
2.1.2. Définitions de la blockchain	10
2.2 Types de blockchain	11
2.3 Architecture de la blockchain	14
2.3.1. Structure de la blockchain	15
2.3.2. Fonctionnement de la blockchain	17
2.4 Protocoles de consensus	19
2.5 Smart contracts (contrats intelligents)	21
2.6 Exemple de blockchain : Ethereum	23
CHAPITRE 3 : ÉTAT DE L'ART SUR L'AUTHENTIFICATION DES DOCUMENTS À L'AIDE DE LA BLOCKCHAIN	33
3.1 Authentification de documents	33
3.2 Méthodes d'authentification de documents	33
3.3 Travaux existants sur l'authentification de documents à l'aide de la blockchain	33
3.4 Discussion	36

CHAPITRE 4 : APPROCHE D’AUTHENTIFICATION DE DOCUMENTS À L’AIDE DE LA BLOCKCHAIN.....	40
CHAPITRE 5 : IMPLÉMENTATION DE L’APPROCHE.....	42
5.1 Protocole d’implémentation.....	42
5.2 Présentation de la solution.....	42
5.3 Discussion des résultats.....	42
CONCLUSION ET PERSPECTIVES	44
RÉFÉRENCES	45
ANNEXES	48

LISTE DES FIGURES

Figure 1 : Projet de chronogramme des travaux	6
Figure 2 : Réseau basé sur les Serveurs vs Réseau P2P	15
Figure 3 : Exemple d'entête d'un bloc.....	16
Figure 4 : Schéma simplifié d'une chaîne de blocs	16
Figure 5 : Mécanisme de fonctionnement global de la blockchain [14].....	18

LISTE DES TABLEAUX

LISTE DES SIGLES ET ABREVIATIONS

CSS	Cascading Style Sheets ou Feuilles de Style en Cascade
HTML	HyperText Markup Language ou Langage de balises pour l'hypertexte
HTTP	Hypertext Transfer Protocol ou Protocole de Transfert Hypertexte
IBAM	Institut Burkinabè des Arts et Métiers
MC	Maître de conférences
P2P	Peer-to-Peer
PoA	Proof of Authority ou Preuve d'Autorité
PoS	Proof of Stake ou Preuve d'Enjeu
PoW	Proof of Work ou Preuve de Travail
SHA-256	Secure Hash Algorithm (algorithme de hachage sécurisé) 256 bits
IDE	Integrated Development Environment (environnement de développement intégré)
API	Application Programming Interface (interface de programmation d'application)
JSON	JavaScript Object Notation

CHAPITRE 1 :

INTRODUCTION GENERALE

CHAPITRE 1 : INTRODUCTION GENERALE

La blockchain est une révolution technologique en plein essor au cours de ces dernières années. Reconnue pour ses propriétés de non-répudiation, de transparence, et de stockage décentralisée, elle présente de nombreux avantages dans plusieurs aspects de la vie humaine, et s'avère un moyen efficace pour améliorer les services publics gouvernementaux en particulier. En effet, dans le domaine des services gouvernementaux, il est de nos jours très récurrent de retrouver sur la place publique numérique, des projets de documents administratifs et même des documents administratifs falsifiés, causant ainsi beaucoup de dommages. Malheureusement au Burkina Faso, ce phénomène prend une allure inquiétante. De ce fait, et sachant les possibilités qu'offre la blockchain, nous pensons que l'utilisation de cette technologie dans les processus d'authentification de documents administratifs, pourrait apporter plus de facilité, de fiabilité et de sécurité.

Dans ce chapitre, nous présentons le contexte général dans lequel est né le sujet, et le problème que l'on se propose de résoudre. En plus d'énoncer des hypothèses de recherche, nous y présentons les objectifs et les résultats attendus du sujet. Ce chapitre fournit également une vue d'ensemble du déroulement de notre étude.

1.1. Contexte et justification

La fiabilité de certains documents administratifs est de plus en plus controversée. Cela pourrait être dû en partie, au développement et à l'exploitation malsaine des multiples outils basés par exemple sur l'Intelligence Artificielle (IA). En effet, ce phénomène est caractérisé par le fait que de nombreux cas réels de faux « documents administratifs » ont été retrouvés sur des espaces numériques publics, mettant en déroute bon nombre de citoyens. Parmi ces cas, la majorité a nécessité des démentis officiels venant des structures étatiques que nous qualifions de « victimes » de faux. Par exemple, nous avons constaté entre autres que :

- le 23 janvier 2025, le Ministère de l'Économie, des Finances et de la Prospective, à travers sa page Facebook, a alerté le public comme suit : « *Des avis au public, faussement attribués au ministère de l'Economie et des Finances faisant état de l'ouverture de sessions d'investissement sur les cryptomonnaies sont diffusés sur les réseaux, aux fins de spoliation des citoyens. Le ministère de l'Economie et des Finances tient à rassurer l'opinion qu'il n'est nullement associé à cette initiative qui n'est autre que de l'arnaque et invite les*

citoyens à la vigilance et à dénoncer les auteurs et les complices de telles pratiques auprès des autorités compétentes. » [1].

- le 16 août 2024, le Ministère de la Fonction Publique, du Travail et de la Protection Sociale publiait un démenti, signé DCRP/MFPTPS, sur sa page Facebook en ces termes « *Une loi portant statut général des agents publics et un projet de loi portant statut général des agents publics circulant sur les réseaux sociaux sont faux. Ces textes ne proviennent pas des services techniques du Ministère de la Fonction Publique, du Travail et de la Protection Sociale, ni du Gouvernement ou de l'Assemblée législative de Transition* » [2].
- le 18 janvier 2024, le Service de Communication et des Relations Publiques de la Direction Générale des Douanes a, de même, publié un démenti sur sa page Facebook comme suit : « *Le Service de Communication et des Relations Publiques de la Direction Générale des Douanes informe le public que les communiqués, ci-dessous, sur une supposée vente aux enchères de véhicules n'émanent nullement des services des Douanes* » [3].
- le 06 octobre 2023, le Ministère des Affaires Etrangères du Burkina Faso, à travers la DRCP/MAECR-BE, avait également publié sur sa page Facebook, ce qui suit : « *Depuis un certains temps un communiqué relatif à une bourse canadienne et impliquant le Ministère des Affaires Etrangères du Burkina Faso circule sur les réseaux sociaux. Le Ministère des Affaires Etrangères du Burkina Faso, apporte un démenti à ce communiqué qui est certainement l'œuvre d'individus mal intentionnés.* » [4].
- le 30 novembre 2021, l'Institut national de la statistique et de la démographie (INSD) avait aussi démenti sur sa page Facebook, un faux recrutement d'étudiants dont il serait l'auteur [5].

Le plus souvent, une copie de chaque document en question a été marquée d'insigne de faux par les structures « victimes », puis annexée aux différents démentis. De ce fait, ces documents pourraient être classés « administratifs », car par définition, peut être vu comme document administratif, tout acte produit ou reçu, dans le cadre de la mission de service public, par l'Etat, les collectivités territoriales ainsi que par les autres personnes de droit public ou les personnes de droit privé chargées d'une telle mission [6]. Il peut s'agir d'un communiqué, un certificat (de prise /cessation de service, administratif, ...), un diplôme, un extrait d'acte de naissance, etc.

Ainsi, au regard de la fréquence progressive et élevée de la falsification et/ou fraude de ces types de document, les citoyens, et même des acteurs de l'Administration posent de nombreuses

préoccupations sur l'authenticité, ou l'originalité de tel ou tel acte (document) administratif qui se présente à eux.

A l'image de l'IA, la Blockchain est une technologie novatrice en pleine croissance. C'est une technologie de registre numérique distribué qui s'applique dans divers domaines. En outre, la blockchain étant immuable, décentralisée et transparente, pourrait être une alternative aux préoccupations susmentionnées.

C'est pourquoi, dans ce contexte générale, nous nous intéressons donc à l'utilisation de la technologie blockchain, notamment ses protocoles de consensus, de vérification et de validation, pour résoudre la question d'authentification des documents administratifs, afin de réduire les risques de fraudes et de falsifications de ces documents. D'où le thème du présent mémoire « *Authentification de documents administratifs à l'aide de la blockchain* » que nous nous proposons d'étudier.

1.2. Problématique et hypothèses

La falsification (ou fraude) documentaire est un problème d'actualité auquel est confrontée particulièrement l'administration publique. Afin de conserver leur image, d'assurer une bonne gouvernance et d'éviter l'usage du faux, les structures publiques et privées s'efforcent de vérifier elles-mêmes, manuellement les dossiers des usagers/clients. A défaut, elles délèguent et suivent une longue procédure de vérifications de ces dossiers par des tiers, moyennant des ressources (financières, humaines, temps, ...) considérables. Elles sont également contraintes de rester plus ou moins en veille constante pour démentir d'éventuels faux documents à elles attribués. Cela diminue non seulement le temps consacré aux activités règlementaires des services administratifs, mais augmente les coûts de contrôle et les délais de prestations de services.

De ce fait, quel outil ou quelle technologie peut être mise en œuvre pour faire face à cette situation ? pour nous, la problématique qui se dégage, c'est comment la technologie blockchain peut-elle garantir l'authenticité des documents administratifs de façon plus sécurisée, transparente et efficace ? Autrement dit, comment la blockchain peut-elle aider les administrations à prévenir et/ou à détecter les tentatives de falsification de documents administratifs ? Cette technologie, peut-elle permettre à un service public destinataire d'un document administratif de savoir si oui ou non, il s'agit bien d'un document authentique ?

Pour traiter cette problématique, nous posons les hypothèses suivantes :

Hypothèse 1 : l'intégration de la blockchain dans une solution d'authentification de documents permet de renforcer la sécurité et l'intégrité des documents administratifs.

Hypothèse 2 : la blockchain permet de vérifier l'authenticité en temps réel des documents administratifs.

Hypothèse 3 : l'utilisation de la blockchain dans le processus d'authentification de documents diminue les cas de falsifications et fraudes des documents administratifs.

Pour tester ou mesurer ces hypothèses, nous recourons à des variables tels que le temps moyen de vérification des documents administratifs avant et après l'implémentation de notre solution, et le niveau de satisfaction d'un échantillon de services administratifs. L'on pourra également s'appuyer sur le taux d'incidents de sécurité liés à la manipulation ou à la falsification des documents administratifs sur une période donnée.

1.3. Objectif du sujet

L'objectif principal de notre sujet est de proposer une approche d'authentification de documents administratifs à l'aide de la technologie blockchain.

De façon spécifique, il s'agit pour nous, de :

- faire une revue de littérature de la technologie blockchain et son utilisation dans les processus d'authentification des documents ;
- proposer une approche d'authentification de documents administratifs à l'aide de la blockchain ;
- implémenter une plateforme numérique basée sur l'approche et qui offre la possibilité de vérifier l'authenticité de documents administratifs.

1.4. Résultats attendus

Les travaux devront aboutir à des résultats qui présentent la manière dont les structures pourront utiliser la blockchain pour résoudre, entre autres, leurs problèmes de falsification et de fraude documentaire. Concrètement, les résultats suivants sont attendus au terme de notre étude :

- une revue de littérature de la technologie blockchain et son utilisation dans les processus d'authentification des documents est faite ;
- une approche d'authentification de documents administratifs à l'aide de la blockchain est proposée ;
- une plateforme numérique basée sur l'approche et qui offre la possibilité de vérifier l'authenticité de documents administratifs est implémentée.

1.5. Organisation du travail

Le présent mémoire est l'aboutissement de plusieurs travaux suivant un projet de chronogramme spécifique illustré par la **figure 1** ci-dessous.

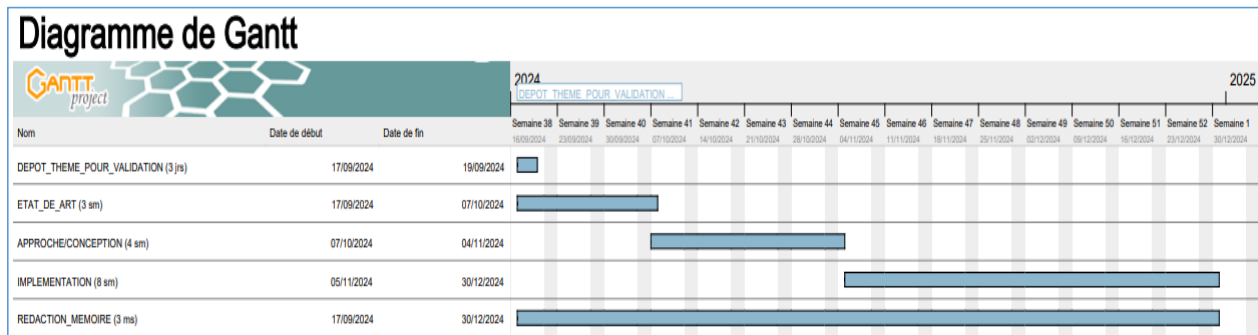


Figure 1 : Projet de chronogramme des travaux

Ce mémoire est organisé en cinq (05) principaux chapitres. En dehors de ce présent **chapitre 1**, le reste du présent mémoire est structuré comme suit :

- **le chapitre 2 est intitulé « technologie blockchain »**. Il est consacré à l'historique et à la définition de la blockchain et ses concepts dérivés. En plus des types de blockchain, nous y présentons l'architecture de la blockchain, les contrats intelligents et les protocoles de consensus. Nous y présentons particulièrement un exemple de blockchain populaire, à savoir Ethereum.
- **le chapitre 3 a pour titre « état de l'art sur l'authentification des documents à l'aide de la blockchain »**. Ce chapitre fait une synthèse des connaissances sur l'authentification de documents, les méthodes existantes d'authentification de documents, et les travaux existants sur les processus d'authentification de documents à l'aide de la blockchain. Nous menons, à la suite de cette synthèse, une discussion sur les travaux existants.
- **le chapitre 4 « approche d'authentification de documents à l'aide de la blockchain »** est dédié à la présentation de notre approche de résolution de la problématique. Nous y listons les étapes à suivre pour la mise en œuvre de notre solution. Nous y fixons aussi le type de document administratif sur lequel nous travaillons.
- **le chapitre 5 « implémentation de l'approche »** est réservé à la mise en œuvre concrète de notre approche. Nous y présentons le protocole d'implémentation à savoir l'environnement d'implémentation de la solution, les éléments de conception, les outils et technologies, etc. C'est également le lieu de présenter la solution développée et faire une discussion sur les résultats obtenus.

- **la conclusion** générale dans laquelle nous dressons un bilan des principales actions réalisées dans le cadre de nos recherches sur le sujet. Ce bilan est consolidé de perspectives. Le présent mémoire prend fin avec la présentation des références bibliographiques/webographiques et des annexes.

Au terme de ce premier chapitre, nous avons décrit le contexte de notre sujet de recherches « *Authentification de documents administratifs à l'aide de la blockchain* ». Nous y avons également énoncé la problématique à résoudre, tout en formulant les hypothèses et les objectifs de recherche. Dans le chapitre suivant, nous faisons un état des connaissances sur la technologie blockchain.

CHAPITRE 2 :

TECHNOLOGIE BLOCKCHAIN

CHAPITRE 2 : TECHNOLOGIE BLOCKCHAIN

On ne peut parler de Blockchain sans aborder la notion de transaction. La transaction est une opération d'échange qui implique plusieurs parties [7]. Dans ce sens, l'opération peut être commerciale ou boursière, un contrat ou accord, etc. Parmi les parties impliquées dans une transaction, occupe en bonne position un tiers de confiance qui permet d'une part de sécuriser la transaction et d'autre part de certifier la validité de ladite transaction. En effet, le tiers de confiance est une entité neutre et indépendante telle qu'une institution financière ou un notaire.

La mondialisation de ce système de transaction à partir des années 1960 a fait augmenter de manière fulgurante, du nombre de transactions. Cela a entraîné l'accroissement des risques liés à l'authenticité des transactions, qui, jusque-là étaient transcrites dans des registres (document) physiques. Mais grâce à l'avènement et à l'évolution rapide de la Technologie, notamment dans les domaines du web et de la cryptographie, le principe du tiers de confiance disparaît progressivement au profit de ce qu'il convient d'appeler « Blockchain ».

Dans ce chapitre, nous faisons une revue de littérature sur la technologie blockchain. En plus d'aborder les origines et définitions de la blockchain, nous présentons sa classification et son architecture. Nous y parlons également des différents mécanismes de consensus, des contrats intelligents et de la blockchain Ethereum en particulier.

2.1 Historique et définitions de la blockchain

2.1.1. Historique de la blockchain

Dès les années 1990, la perspective de digitalisation des documents sous format numérique soulevait déjà la question de savoir comment certifier la date à laquelle un document a été créé ou modifié pour la dernière fois. Comment faire en sorte qu'un utilisateur ne puisse pas antedater ou modifier la date d'un document mis sur support numérique ? dans [8], *Stuart Haber et W. Scott Stornetta* ont proposé dans ce sens, des procédures informatiques pratiques pour l'horodatage numérique de documents sous forme numérique. Leurs réflexions sur entre autres le hachage, et les signatures numériques, les ont permis d'incorporer, en 1992, le concept d'arbre de Merkle au système d'horodatage de documents avec le concours de *Dave Bayer*. Cette innovation a amélioré l'efficacité du système en permettant à plusieurs documents d'être regroupés en un seul bloc [9].

Dans [10], le chercheur *Ittai Abraham* a affirmé : « *The longest running blockchain started in 1995 and is still running strong today. (...)* » ; ceci pour indiquer que le premier système de certification

décentralisé est celui de la société Surety, qui publie chaque semaine depuis 1995 un certificat cryptographique de sa base de données dans la rubrique « Annonces et objets trouvés » du « New York Times ». Pour en venir, le concept de Blockchain en lui-même, basé sur la cryptographie, a été évoqué pour la première fois au début des années 1980. Mais de nos jours, il est impossible de dissocier les concepts de blockchain et de crypto-monnaies car elles constituent le point essentiel d'émergence de la blockchain. En effet, la crypto-monnaie est une monnaie virtuelle dont l'implémentation repose sur des algorithmes cryptographiques permettant de générer de la monnaie et de faire des transactions anonymes entre des paires sur internet. Dans ce sens, le bitcoin, une crypto-monnaie qui s'est rapidement imposée de manière non triviale, a été annoncé en 2008 par son mystérieux – mystérieux, car « Satoshi Nakamoto » est largement considéré comme un pseudonyme, et la véritable identité de l'inventeur du bitcoin reste une inconnue – développeur, **Satoshi Nakamoto** [11]. Le bitcoin, selon *S. Nakamoto* dans [12] (plus détaillé par *G. Ferréol et R. Romain* dans [13]), repose sur trois fondamentaux à savoir : le réseau pair-à-pair sans autorité centrale, les transactions et le triple protocole de vérification-consensus-validation. Ces éléments constituent une chaîne de blocs (ou blockchain en anglais).

2.1.2. Définitions de la blockchain

La blockchain se définit de plusieurs manières.

De manière basique, la blockchain est une technologie numérique de stockage chronologique et de transmission d'informations sous forme de blocs reliés les uns aux autres de manière sécurisée et sans autorité centrale. En termes plus simple, le Mathématicien *Jean-Paul Delahaye*, la définit comme étant : « *un très grand cahier, que tout le monde peut lire librement et gratuitement, sur lequel tout le monde peut écrire, mais qui est impossible à effacer et indestructible* ».

De manière technique, la blockchain est assimilable à une base de données distribuée, dont les informations envoyées par les utilisateurs et les liens internes à la base sont vérifiés, puis groupés à intervalles de temps réguliers (environ 10 minutes dans le cas de bitcoin) en blocs. L'ensemble de ces blocs est sécurisé par cryptographie et forme ainsi une chaîne de plus en plus longue. Par extension, une chaîne de blocs est une base de données distribuée qui gère une liste d'enregistrements théoriquement protégés contre la falsification ou la modification par les nœuds de stockage ; c'est donc un registre distribué et sécurisé de toutes les transactions effectuées depuis la création de la transaction initiale [14].

On peut donc comprendre que cette technologie est basée sur le concept de grand livre distribué ou de base de données partagée. Cela implique que dans le réseau blockchain, chaque participant (nœud) au réseau a sa propre copie de la base de données. Pour y parvenir, **un mécanisme (ou algorithme) de consensus** est utilisé. En effet, le mécanisme de consensus est essentiellement caractérisé par :

- **un accord unanime sur le contenu des données**, afin de permettre à tous les nœuds (ou du moins la majorité des participants) du réseau de valider et de s'accorder sur quelles données doivent être inscrites dans la blockchain, du fait de la présence de données contradictoires. La cryptographie est utilisée lors de la validation des transactions.
- **une conformité des copies des données convenues**, afin de s'assurer que toutes les transactions ajoutées à la blockchain sont les mêmes pour chaque utilisateur.
- **une absence de tricherie par altérations des données ou tentative de fraude**, qui est garanti grâce à des mécanismes cryptographiques qui rendent toute tentative de modification ultérieure des données pratiquement impossible. En effet, dans une chaîne de blocs, les transactions sont horodatées en permanence. Cette sorte d'archivage empêche la suppression ou l'inversion des transactions une fois ajoutées à la chaîne de blocs, et dès que d'autres blocs ont été ajoutés à la suite.

Tout se passe dans un réseau distribué de sorte à ce que si un nœud tombe en panne, les autres peuvent continuer à fonctionner de façon transparente ; ce qui garantit la disponibilité et la fiabilité dans les transactions. On parle précisément de système distribué décentralisé. Dans ce système, chaque participant peut vérifier les informations de manière indépendante car les processus de vérification ne dépendent d'aucune autorité centralisée [15].

La technologie blockchain, dans son évolution, se distingue en différentes formes que nous trouvons essentiel de présenter.

2.2 Types de blockchain

La classification de la technologie blockchain est possible du fait de son évolution générationnelle. Dans [16], *Imran Bashir* discute de quatre (04) générations (ou niveaux) de la blockchain. Il s'agit de la :

- Blockchain 1.0 : cette génération ne concernait que les crypto-monnaies car elle a été introduite avec l'invention du Bitcoin. Elle inclut donc les applications de base telles que les paiements et les applications servant à effectuer de simples transferts de valeurs.
- Blockchain 2.0 : elle est une évolution de la première génération à travers l'intégration des contrats intelligents et autres applications dérivées des services financiers.
- Blockchain 3.0 : les blockchains de la troisième génération ont permis d'envisager, au-delà de l'industrie des services financiers, de nombreuses autres applications à usage général telles que les médias, la santé, le gouvernement, etc.
- Blockchain X : la génération X permet de se projeter dans une vision où la blockchain va fournir des services dans tous les domaines de la société.

Outre cette évolution générationnelle, il existe une classification de la blockchain basée sur le réseau. Les principaux types de réseau blockchain sont les blockchains publiques, les blockchains privées et les blockchains de consortium.

Les blockchains publiques

Ces blockchains sont également appelés « blockchains sans permission ». C'est des blockchains qui sont ouvertes au public – donc accessibles à tous, au point de ne requérir aucune permission spécifique à l'entrée, ni au moment de réaliser une transaction – et chaque utilisateur peut conserver sans autorisation préalable, une copie du registre sur son nœud local. Les blockchains publiques sont aussi caractérisées par le fait que toute personne, en tant que nœud du réseau distribué, peut participer au processus de prise de décision. La prise de décision sur l'état de ce type de blockchain nécessite l'utilisation d'un mécanisme de consensus distribué. Il peut arriver qu'un nœud participant soit récompensé pour sa participation. Ce type de blockchain est en général open source. Ethereum et Bitcoin¹ sont des exemples de blockchains publiques [17].

Les blockchains privées

Les blockchains privées ne sont pas ouvertes au public. Elles sont exclusivement accessibles sur invitation. Tous les membres participants de ce réseau blockchain se connaissent et se font confiance. Les membres participants peuvent être un groupe d'individus ou d'organisations qui ont décidé de partager le registre entre eux. Dans ce type de blockchain, un mécanisme de consensus

¹ Détails sur : <https://www.blockchain.com/fr/explorer>

est utilisé pour valider l'écriture des données parmi ses participants privilégiés. Par exemple, cette approche est plus appropriée lorsque plusieurs succursales d'une même entreprise (organisation) décident de l'utiliser pour se partager directement des informations. Aussi appelées blockchains permissionnées ou blockchains à autorisation, Hyperledger et Ripple sont des exemples de blockchain privée fréquemment cités [17].

Les blockchains semi-privées

Encore appelé blockchain de consortium, ce type de blockchain constitue un mixage entre la blockchain publique et la blockchain privée.

Dans ce type de blockchain, seuls quelques nœuds sélectionnés sont prédéterminés à se partager la responsabilité de la maintenance et de la sécurisation du réseau blockchain. Ils ont la responsabilité de déterminer les droits d'accès aux données. Les nœuds participants, eux, sont invités. Les décisions sont prises par la majorité des acteurs présélectionnés. Cela signifie qu'en dehors des données spécifiques stockées et contrôlées dans la blockchain, le reste des données sont accessibles au public. Ainsi, des membres publics peuvent vérifier (à l'aide de contrats intelligents) si les transactions privées ont été effectuées. Le fait d'avoir des droits de lecture pouvant être publics ou limités aux participants permet de préserver la confidentialité des données, comme dans les blockchains privées. BigchainDB, EEA et R3 sont des exemples de blockchain de consortium [18].

Outre ces trois (03) principaux types de blockchain selon les attributs réseau, il existe des blockchains dérivées telles que les sidechains (chaines secondaires ou chaîne de transactions gérée par une sous-communauté) [7], les grands livres autorisés, les grands livres distribués, les grands livres partagés, les blockchains entièrement privées et propriétaires, les blockchains à jetons, les blockchains sans jetons, etc. [16]. Nous nous intéressons aux grands livres autorisés et aux blockchains entièrement privées et propriétaires.

Un grand livre autorisé est une blockchain dans laquelle l'utilisation d'un mécanisme de consensus distribué n'est pas nécessaire, car les utilisateurs sont connus et se font confiance. Les participants au réseau du grand livre autorisé peuvent utiliser un protocole d'accord pour maintenir une version partagée de la vérité sur l'état des enregistrements dans la blockchain. Dans ce cas, il n'est pas nécessaire que le grand livre autorisé soit privée, car elle peut être publique avec un

contrôle d'accès réglementé. Les grands livres autorisés sont aussi appelés blockchains ou registres avec permission.

Comme leur nom l'indique, les **blockchains entièrement privées et propriétaires** ne sont pas ouvertes au grand public. Mais, dans des contextes privés spécifiques au sein d'une organisation, il pourrait être nécessaire de partager des données et de fournir un certain niveau de garantie quant à l'authenticité des données. Ces blockchains pourraient être utiles, par exemple, pour la collaboration et le partage de données entre différents départements gouvernementaux [16].

Bien qu'il y ait plusieurs types de blockchains, la structure et le mode fonctionnement de l'ensemble des blockchains qui permettent de garantir la sécurité des transactions restent quasiment les mêmes.

2.3 Architecture de la blockchain

Pour une meilleure compréhension de cette section, il semble indispensable de revenir sur les termes « hash », « arbre/racine de Merkle », et « mineur » qui sont des composants fondamentaux de la technologie blockchain.

Un **hash** est le résultat d'une fonction mathématique (SHA-256, Ethash, MD5, etc.) qui permet d'obtenir une empreinte numérique unique de taille fixe à partir d'une donnée d'entrée (fichier, texte, ...) de taille non bornée. Par exemple, l'algorithme SHA-256 (32 octets) produit toujours une valeur de sortie hexadécimale de 64 caractères. En cryptographie, l'empreinte numérique générée est difficilement (voire impossible) devinable par un humain. De plus, la fonction de hachage est irréversible en ce sens qu'on ne peut pas retrouver la donnée d'origine à partir du hash.

L'**arbre de Merkle ou arbre de Hash (ou hachage)** a été créé en 1979 par le cryptographe informaticien Ralph Merkle. Il est beaucoup utilisé dans la blockchain et la cryptographie [19]. C'est une structure de stockage de données sous forme d'arbre binaire où chaque nœud de l'arbre est identifié par un hash. En effet, les nœuds initiaux (nœuds enfants ou feuilles) sont associés à un nœud supérieur appelé nœud parent. Ainsi, un nœud parent a un identifiant unique résultant du hachage de ses deux (02) nœuds enfants. Cette structure est répétée jusqu'au nœud racine ou racine Merkle (Merkle root), dont l'empreinte est associée à tous les nœuds de l'arbre. Cela permet de rechercher efficacement des transactions dans la chaîne de bloc et d'empêcher les falsifications [20]. L'annexe 1 présente une illustration de l'arbre de Merkle où les hashes sont nécessairement

couplés par paire de nœud – le nœud EF y étant seul et différent de la racine, a été dupliqué et couplé avec lui-même.

Un **mineur** est un nœud (ordinateur doté de grande puissance de calcul) actif du réseau de blockchain qui sélectionne des transactions, les vérifie et participe à leur validation. Il se distingue des autres nœuds du réseau tels que les clients SPV (Simple Paiement Verification) et les clients Web [20]. Aussi, il convient de rappeler que *“Tous les mineurs sont des nœuds, mais tous les nœuds ne sont pas nécessairement des mineurs.”* P.73 [20].

2.3.1. Structure de la blockchain

La blockchain, comme son nom l’indique, est une chaîne ou liste chaînée reliant des **blocs en retour** et hébergée dans des nœuds en réseau. Un bloc regroupe plusieurs transactions.

En effet, la structure de la blockchain repose sur l’architecture réseau distribué Peer to Peer (P2P) – aussi appelé pair-à-pair en français – qui est un réseau d’égal à égal. Ce type de réseau regroupe un ensemble d’ordinateurs appelés nœuds qui partagent les informations ou fichiers entre eux de manière directe, rapide et abordable. Ces nœuds contiennent chacun, une copie de la blockchain et fournissent un consensus sur l’état de celle-ci à tout moment. La **figure 2** ci-dessous présente un schéma de réseau décentralisé P2P où chaque utilisateur ou nœud possède à la fois le rôle de serveur et de client ; ce qui est différent pour les réseaux classiques.

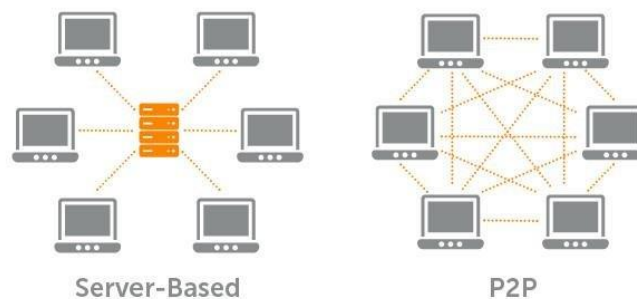


Figure 2 : Réseau basé sur les Serveurs vs Réseau P2P

Source figure : https://www.researchgate.net/figure/Reseau-base-sur-les-Serveurs-vs-Reseau-P2P_fig5_335174496

Dans une blockchain, les blocs sont plus ou moins importants en fonction du nombre de données qu’ils renferment. En effet, chaque bloc contient deux (02) parties à savoir l’entête (header) et le corps (facts) du bloc.

Le header contient plusieurs informations clés telles que [21] :

- **la version** qui indique le protocole de validation des règles ;
- **le hash du bloc précédent** qui assure liaison entre les blocs afin de constituer la chaîne ;
- **le hash de la racine de Merkle** qui synthétise les informations que renferment toutes les transactions du bloc ;
- **le timestamp (date et heure de création)** pour l'horodatage du bloc qui précise le temps de minage ;
- **les bits** qui indique la valeur actuelle de la difficulté de minage ;
- **le nonce** qui est un numéro aléatoire utilisé lors du minage pour trouver un hash valide.

En pseudo-code, une entête d'un bloc peut ressembler au contenu de la **figure 3** ci-dessous.

```

1  BlocHeader: {
2      · Version: 1,
3      · PreviousBlockHash: "00000000000004X8G...",
4      · MerkleRoot: "3a5bc234ad...",
5      · Time: 1234567890,
6      · Bits: 1703ddf8c3,
7      · Nonce: 2085406893
8  }
```

Figure 3 : Exemple d'entête d'un bloc

Le corps du bloc contient les transactions qui doivent être stockés dans les bases de données. Ces transactions sont appelées « facts » ou « faits ». La transaction est l'élément de base de la blockchain Bitcoin (primitive des autres blockchains). La **figure 4** ci-dessous présente un exemple simplifié d'une chaîne de blocs.

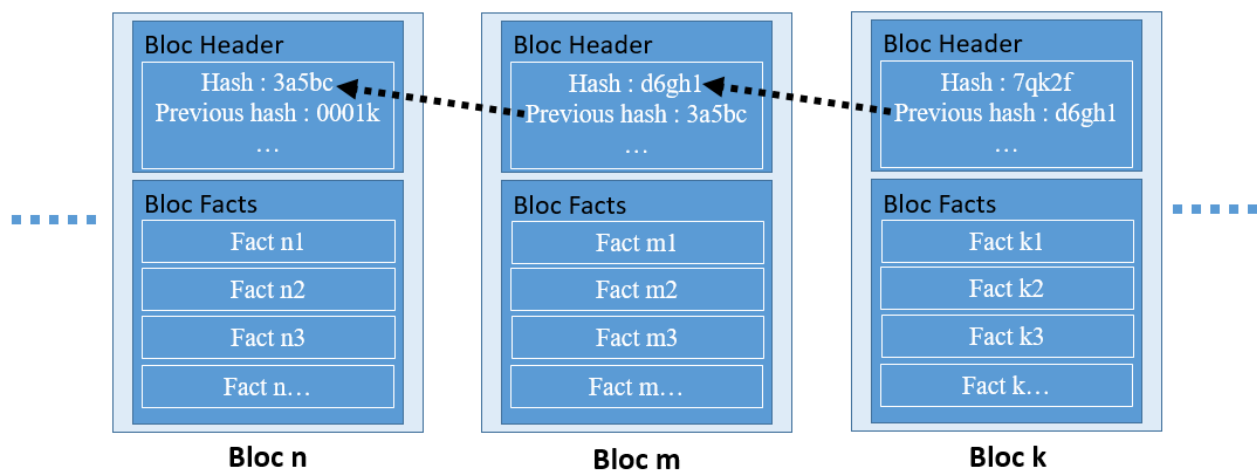


Figure 4 : Schéma simplifié d'une chaîne de blocs

Dans ce schéma, nous pouvons constater que **bloc_m.previous_hash = bloc_n.hash** et **bloc_k.previous_hash = bloc_m.hash** ; ce qui justifie la notion de blocs chaînés « en retour ». Cependant, il peut arriver qu'il y ait des chaînes de blocs orphelines (chaînes secondaires). Dans

ce cas, la chaîne principale est composée de la plus longue suite de blocs après le bloc initial (ou bloc de genèse).

En générale, les facts sont organisés de manière séquentielle, de la plus ancienne à la plus récente, et peuvent être des transactions monétaires, des données médicales, des informations industrielles, des logs systèmes, etc. [20].

La logique de chaînage des blocs peut être décrite comme suit :

Soient B_0 , B_1 , B_2 , les blocs représentant respectivement les bloc n, bloc m et bloc k de la **figure 4** ci-dessus ; où B_0 est supposé bloc de genèse, B_1 le bloc fils de B_0 et B_2 le petit fils de B_0 et fils de B_1 .

Chaque bloc de la chaîne est identifié de manière unique par un hash obtenu en SHA-256 (pour Bitcoin), Ethash (pour Ethereum), etc. Par exemple, la donnée (ou le texte) d'entrée $\alpha = \textit{Exemple de hash d'un bloc dans une chaîne de blocs}$ a comme valeur de hash SHA-256, la sortie $\beta = 16958df5ae0040030217620a52f49e4398588cbbfa1f8bc1ef751fd2ba384ba5$. Et la moindre modification de α engendre obligatoirement un changement de β . Chaque bloc fait référence au bloc précédant à travers le hash de celui-ci. En effet, le hash de B_0 est référencé (ou inscrit comme `previous_hash`) dans l'entête de B_1 et celui de B_1 dans l'entête de B_2 , formant ainsi une chaîne.

Dans une chaîne donnée, si l'identité du bloc de genèse ou d'un bloc parent change, l'identité des blocs enfants changera obligatoirement. Autrement dit, si un utilisateur modifie B_0 , cette modification entraînera un changement du hash de B_0 . Ce changement du hash de B_0 imposera un changement du pointeur « `previous_hash` » dans B_1 ; ce qui entraînera un changement du hash de B_1 , qui, à son tour changera le hash de B_2 , et ainsi de suite. De ce fait, cette opération de cascade implique le recalcul des hash de tous les blocs suivants dès qu'un bloc parent (ayant plusieurs descendants) viendrait à être modifier. Et plus la chaîne de blocs est longue, plus le recalcul devient énorme et coûteux, et plus l'historique (horodatage via la clé **timestamp** ou **time** du header) devient profond. Ceci explique le caractère immuable de la blockchain. Voir une simulation sur [22].

En sus de cette structuration, comment fonctionne la technologie blockchain ?

2.3.2. Fonctionnement de la blockchain

Le mécanisme global de fonctionnement de la blockchain passe par l'initiation d'une transaction, la validation de bloc via un consensus, et l'ajout du bloc validé à la chaîne précédente.

Dans [20], *Oussama* fait une présentation de ce mécanisme sur laquelle nous nous appuyons ici à travers la **figure 5** ci-dessous.

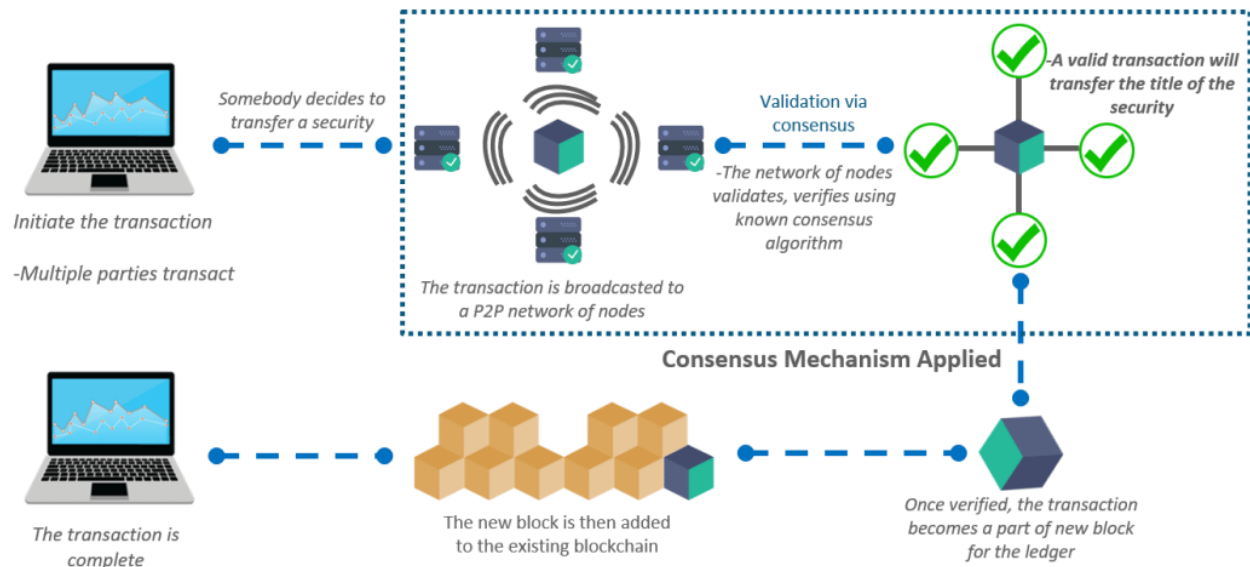


Figure 5 : Mécanisme de fonctionnement global de la blockchain [20]

En effet, le mécanisme de fonctionnement est résumé en six (6) étapes à savoir :

- 1- Un utilisateur de la blockchain initie une transaction Tx .
- 2- Tx est publiée (ou diffusée) sur le réseau de blockchain.
- 3- Les mineurs entrent en compétition et valident Tx . Dans ce cas, une empreinte digitale (hash) est générée et associée à Tx .
- 4- Si Tx est vérifiée et validée, elle est ajoutée à d'autres transactions dans un bloc en construction. Ce processus permet de garantir que toutes les transactions sont légitimes. Légitimes pour indiquer que les transactions sont authentiques, valides (respect des tailles du bloc), non falsifiées, et acceptées par le réseau.
- 5- Après ce mécanisme de consensus, le bloc, désormais construit, est ajouté à la chaîne de blocs précédents (blockchain existante). Avant qu'un bloc ne soit ajouté à la chaîne existante, il est diffusé à tous les nœuds du réseau blockchain, suivi de l'acceptation de sa validité par ceux-ci. Ce bloc devient alors sécurisé et inaltérable. Cependant, si un bloc n'est pas entièrement validé, il ne peut pas être ajouté à la chaîne. Concernant les mécanismes de consensus, les plus populaires et les plus utilisés sont le Proof of Work (PoW) ou le Proof of Stake (PoS) [21].
- 6- Ainsi, Tx est confirmée et considérée comme effectuée avec succès.

La figure présentée dans la section « ANNEXE 2 », permet de suivre l'itinéraire d'une transaction initiée dans un réseau de blockchain.

Dans le fonctionnement global de la blockchain, le concept de mécanisme (ou méthode) de consensus occupe une place importante. Qu'en est-il exactement ?

2.4 Protocoles de consensus

Dans le contexte de blockchain, le concept de consensus est indispensable et constitue l'épine dorsale de la technologie.

En générale, c'est une procédure qui consiste à dégager un accord sans procéder à un vote formel, ce qui évite de faire apparaître les objections et les abstentions [23]. Dit autrement par [24], le consensus désigne toute situation où plusieurs parties se mettent d'accord, sans possibilité d'opposition et sans que les intérêts de l'une ou l'autre des différentes parties ne se trouvent lésés. Le consensus s'établit généralement à l'unanimité, ou tout du moins à la majorité. Le consensus est indissociable du mot voisin « consentement » : il ne revêt pas un caractère irréfutable, il s'agit de quelque chose que l'on admet, sur laquelle on s'accorde, et que l'on accepte comme une vérité ou comme une solution, en réponse à une question ou à un problème donné.

Appliqué à la Blockchain, le consensus est un processus sécurisé par lequel un groupe de pairs (ou nœuds) sur un réseau de blockchain parviennent à un accord unanime pour déterminer quelles transactions de la blockchain sont valides et lesquelles ne le sont pas. Ainsi, il permet de garantir que chaque nouveau bloc ajouté à la chaîne est la seule et unique version de la vérité vérifiée et acceptée par tous les nœuds du système décentralisé et distribué. On parle alors de mécanisme de consensus ou d'algorithme de consensus [25]. D'une blockchain à une autre, la méthode utilisée pour parvenir à cet accord peut certes varier, mais comment parvenir à se mettre d'accord sur la blockchain ?

[26] *Satoshi Nakamoto*, en voulant vérifier l'authenticité d'un réseau blockchain et éviter les doubles dépenses dans le contexte de la crypto-monnaie, a mis en place en 2009, le premier consensus blockchain appelé « consensus de Nakamoto ». L'algorithme de consensus de Nakamoto intègre la tolérance aux pannes byzantines (BFT²) et combine une énigme informatique nécessitant des calculs extrêmement complexes (appelés proof of **work**-PoW ou preuve de travail)

² Détails sur <https://river.com/learn/what-is-the-byzantine-generals-problem/> et <https://crypto.com/glossary/byzantine-fault-tolerance-bft>

afin de dissuader les acteurs malveillants du réseau. Par la suite, le consensus de Nakamoto a évolué en utilisant des ressources « **X** » rares et difficiles à obtenir. En effet, toute PoX incorpore en générale :

- la tolérance aux pannes byzantines afin permettre au réseau de continuer à fonctionner, même en cas de pannes de certains nœuds ou actions malveillantes de ceux-ci ;
- la communication synchrone entre les nœuds de sorte à ce que les messages ou transactions soient livrés dans un délai régulier ;
- une probabilité sur laquelle les nœuds s'accordent sur l'état du réseau ;
- la gouvernance qui indique les nœuds leaders responsables des validations des transactions.

Il existe plusieurs types de protocoles de consensus PoX à savoir : le Proof of Work (PoW), le Proof of Stake (PoS), le Proof of Authority (PoA), le Proof of Activity (PoA), le Proof of History (PoH), le Proof of Importance (PoI), le Proof of Alepsed Time (PoET), le Delegated Proof of Stake (DPoS), le Proof of Capacity/Proof of Space (PoC/PoSpace), et le Proof of Burn (PoB) [25].

En dehors des types de consensus PoX, il existe des consensus classiques basés sur le vote tels que la Tolérance pratique aux pannes byzantines (pBFT), la Tolérance aux pannes byzantines déléguée (dBFT), l'Accord de la Fédération Byzantine (AFB), le Radeau, etc. [26].

Sachant qu'il n'y pas que les types de protocoles de consensus PoX et que chacun de ces protocoles présente des avantages et des inconvénients, nous nous intéressons particulièrement aux PoW, PoS, et PoA qui sont les plus utilisés dans les réseaux blockchain.

La Preuve de travail (Proof of Work – PoW) [25] est le premier algorithme de consensus implémenté dans une crypto-monnaie et utilisé par Bitcoin. Ce mécanisme nécessite une puissance de calcul considérable pour résoudre les problèmes mathématiques complexes et valider un bloc. Cela intervient lorsque parmi plusieurs mineurs (qui se concourent pour gagner la récompense du bloc ou coinbase transactions), chacun s'investi à trouver le Nonce correspondant au hash du bloc candidat à validation. La PoW est efficace en terme de sécurité. Mais son insuffisance est le fait qu'elle exige une quantité importante d'électricité et de ressources matérielles (Application-Specific Integrated Circuits, Graphics Processing Units, Serveurs puissants, Equipements de data center, ...) pour fonctionner.

La preuve d'enjeu (Proof of Stake – PoS), contrairement à la PoW, ne requiert pas une puissance de calcul pour valider les transactions et créer de nouveaux blocs. Dans ce mécanisme, les nœuds validateurs sont sélectionnés en fonction de la quantité de monnaie qu'ils sont prêts à mettre en jeu comme garantie. Le validateur est donc récompensé par des frais de transactions. Ainsi, les nœuds détenteurs de monnaie sont encouragés à agir honnêtement et à sécuriser le réseau de la blockchain (surtout contre les attaques à 51% [27]) au risque de perdre leur mise. Par exemple, Ethereum est passé du PoW au PoS en 2022 [25].

La preuve d'autorité (Proof of Authority – PoA) quant à elle, est un algorithme de consensus approprié pour les blockchains d'entreprise du fait de sa faible consommation en énergie. Elle a été proposé en 2017 par *Gavin Wood*. Ce mécanisme de consensus oblige généralement les utilisateurs de la blockchain à dévoiler leur identité. Ainsi, au lieu de mettre des monnaies en jeu ou de disposer d'une énorme puissance de calcul comme dans les PoS et PoW respectivement, les validateurs mettent leur réputation en jeu pour obtenir le droit de valider les blocs dans la PoA [25]. De ce fait, même si la PoA a l'avantage d'être écologique, elle fait tout de même l'objet de critiques pour le fait qu'elle fonctionne sur le principe de centralisation de droits de validation sur la base de la réputation.

Outre les protocoles de consensus, le concept de contrat intelligent a émergé et fait désormais partie intégrante de la technologie blockchain.

2.5 Smart contracts (contrats intelligents)

Le terme « **smart contracts** » (ou **contrats intelligents** en français) a été utilisé pour la première fois en 1994 par *Nick Szabo* [28]. Il le définit comme "*un ensemble de promesses, spécifiées sous forme numérique, comprenant les protocoles dans le cadre desquels les parties exécutent d'autres promesses*" [29]. En terme simple, un contrat intelligent est un programme informatique implémenté, déployé et exécuté sur une blockchain lorsque certains critères ou spécifications sont remplis. Les contrats sont dits intelligents pour le fait qu'ils s'exécutent de façon autonome lorsque les conditions prédéfinies sont remplies. Leur récente popularité provient de l'émergence des blockchains (même s'ils ne sont pas disponibles dans toutes les blockchains), et notamment d'Ethereum. Les contrats intelligents sont applicables dans plusieurs domaines tels que la santé, les finances, le gouvernement, etc.

Selon [16], l'application des contrats intelligents constitue une caractéristique révolutionnaire de la blockchain pour le fait qu'ils permettent une flexibilité, une autonomisation (par programmation) et un contrôle très souhaitable des actions que les utilisateurs de la blockchain doivent effectuer en fonction de leurs exigences commerciales ou administratives spécifiques.

Techniquement, un contrat intelligent est un code source implémenté dans un langage de programmation de haut niveau tel que Solidity (orienté objet à typage statique), Vyper (de type python bien compatible avec l'EVM), etc., en combinant d'autres outils et technologies web. Ce code est ensuite compilé en bytecode à l'aide de Yul (langage intermédiaire pour compiler du code Solidity) par exemple. Une fois compilé, le contrat est déployé sur une version spécifique d'une machine virtuelle (machine virtuelle Ethereum – EVM) pour y être exécuter. La **figure 6** ci-dessous est un exemple de contrat intelligent écrit en Solidity via l'environnement de développement Remix-Ethereum-IDE³.

```

1  pragma solidity ^0.7.0 ^0.9.0;
2  import "hardhat/console.sol";
3  contract Owner {
4      address private owner; //variable for owner's address
5      event OwnerSet(address indexed oldOwner, address indexed newOwner); // event for EVM logging
6
7      modifier isOwner() { // modifier to check if caller is owner
8          require(msg.sender == owner, "Caller is not owner");
9          _;
10     }
11
12     constructor() { //Set contract deployer as owner
13         console.log("Owner contract deployed by:", msg.sender);
14         owner = msg.sender; // 'msg.sender' is sender of current call, contract deployer for a constructor
15         emit OwnerSet(address(0), owner);
16     }
17
18     function changeOwner(address newOwner) public isOwner { //Change owner by address of new owner
19         require(newOwner != address(0), "New owner should not be the zero address");
20         emit OwnerSet(owner, newOwner);
21         owner = newOwner;
22     }
23
24     function getOwner() external view returns (address) { // address of owner
25         return owner;
26     }
27 }

```

Figure 6 : Exemple d'un contrat intelligent dénommé « Owner »

Les contrats intelligents regorgent d'importants avantages mais également révèlent des insuffisances dans leurs applications. Il s'agit entre autres :

— Avantages [30] :

- **Rapidité, efficacité et exactitude** : le contrat est exécuté immédiatement dès que les conditions (if/when...then...) prédéfinies sont respectées. Aucun temps n'est consacré à rectifier les erreurs.
- **Confiance et transparence** : aucun tiers n'est impliqué dans le contrat entre deux (02) utilisateurs. De plus, les enregistrements chiffrés des transactions sont partagés

³ <https://remix.ethereum.org/>

entre les participants, évitant ainsi la modification des informations à des fins personnelles.

- **Sécurité et économies** : les enregistrements de transaction dans la blockchain sont chiffrés, donc impossible à détourner. De plus, chaque enregistrement est relié aux enregistrements précédents et suivants dans un grand livre distribué. Aussi, l'absence d'intermédiaire pour gérer les transactions conduit à l'absence ou à la réduction des délais et frais associés aux transactions.

— **Inconvénients** [31] :

- **Erreur de l'oracle** : en générale, le contrat intelligent récupère la donnée d'un oracle (source de données) pour s'exécuter. Si cette donnée n'est pas fiable à la source, le contrat va tout de même s'exécuter sans possibilité de revenir en arrière.
- **Piratage** : le contrat est exécuté par un programme informatique qui est lui-même sur la blockchain. Ainsi, si ce programme contient des failles ou des bugs, un hacker qui les découvre, pourrait les utiliser à d'autres fins malsaines.
- **Immuabilité de la blockchain** : la blockchain étant par nature immuable, si le programme a fait une erreur ou s'il a été piraté, il n'est pas possible d'annuler une opération.

Les contrats intelligents sont une illustration parfaite de la possibilité qu'une blockchain soit programmable, contrairement aux premières versions de blockchain. Ethereum que nous présentons dans le point suivant est l'exemple populaire de blockchain programmable pour usage autres que les crypto-monnaies.

2.6 Exemple de blockchain : Ethereum

2.6.1. Historique et évolution d'Ethereum



La blockchain Bitcoin, depuis son déploiement en 2009, est un système décentralisé dédié exclusivement aux paiements digitaux en s'appuyant sur la crypto-monnaie bitcoin. En plus de cette exclusivité, *Vitalik Buterin* (programmeur russo-canadien et co-fondateur de Bitcoin Magazine) a publié en 2013, un livre blanc dans lequel il souligne les autres limitations de Bitcoin et décrit la vision et la conception technique d'Ethereum. En effet, selon *Buterin*, la technologie blockchain pourrait bénéficier de bien d'autres applications que les crypto-monnaies. Cette ainsi que Ethereum a été annoncé lors de la Conférence nord-américaine sur le Bitcoin à Miami en janvier 2014, puis lancé le 30 juillet 2015 [32].

Ethereum, conçue dans le but de lancer des applications décentralisées (dApps), est devenue tout de même la deuxième plus grande blockchain en termes de capitalisation boursière après Bitcoin. Pour être plus précis, il est énoncé dans le livre blanc ceci : *“L'objectif d'Ethereum est de créer un autre protocole pour développer des applications décentralisées, en offrant un ensemble différent de compromis qui sera, nous le pensons, très utile pour une large gamme d'applications décentralisées. Il sera principalement axé sur les situations dans lesquelles le développement rapide, la sécurité des petites applications rarement utilisées et la possibilité pour les différentes applications d'interagir ensemble de façon très efficace sont importants.”* [33]. Cela fait d'Ethereum, une blockchain Turing Complete (peut effectuer tout calcul mathématique, y compris les boucles infinies et conditions) car elle permet à tout développeur d'écrire et d'y déployer des contrats intelligents et des dApps selon ses propres règles et logiques métiers d'entreprise [34]. Et c'est là l'innovation majeure qu'apporte Ethereum par rapport à Bitcoin. Pour rendre le réseau Ethereum plus solide et diversifié, les clients d'exécution (anciennement appelés « clients Eth1 » ou « client Ethereum ») ont été développés en utilisant plusieurs langages de programmations à savoir, Go, Java, Rust, C# et TypeScript [35].

Ethereum, dans son évolution et surtout suite à son piratage à travers la faille du code de The DAO le 17 juin 2016 [32], a connu plusieurs mutations. En effet, la gestion de cette attaque a abouti à la création de deux (02) chaînes de blocs distinctes à savoir l'Ether (ETH) qui est la chaîne officielle et l'Ether classique (ETC) qui ne représente que 15% de la puissance de calcul des mineurs d'Ethereum avant piratage. Cela dit, nos recherches sont orientées vers l'ETH. Outre la scission de la chaîne initiale, Ethereum a été critiqué (au même titre que d'autres blockchain) pour son émission de gaz à effet de serre très élevée (moins écologique). Et pour y remédier, Ethereum a changé son mécanisme de consensus de la preuve de travail (PoW) à la preuve d'enjeu (PoS) le 15 septembre 2022 (Ethereum vers Ethereum 2.0), réduisant ainsi sa consommation énergétique d'environ 99,95% [32].

2.6.2. Architecture et fonctionnement d'Ethereum

A l'image de Bitcoin, Ethereum fonctionne sur un réseau P2P. L'univers d'Ethereum est constitué de plusieurs éléments à savoir le réseau, les mécanismes de consensus, les nœuds et clients, l'EVM (Ethereum Virtual Machine), les comptes, les transactions et blocs, l'ETH (Ether

native), les contrats intelligents et dApps, etc. Ces éléments sont regroupés en composants globaux permettant de schématiser Ethereum de manière simplifiée tel qu'illustre la **figure 7** ci-dessous.

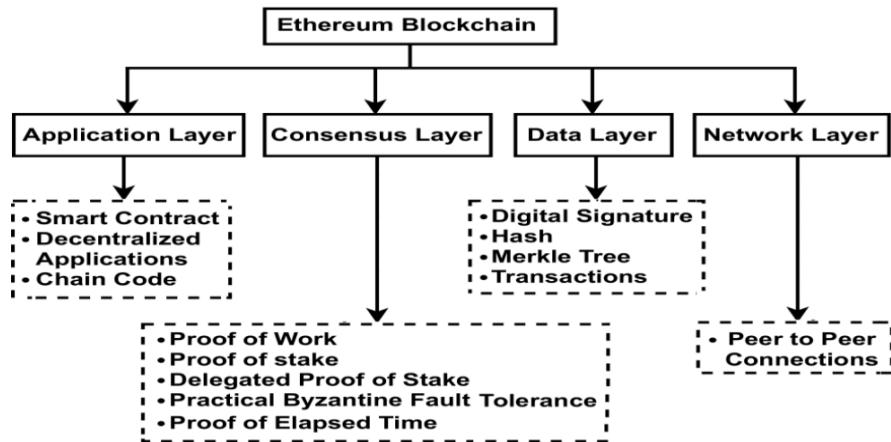


Figure 7 : Architecture en couches de la blockchain Ethereum [34]

Chaque couche de cette architecture joue un rôle spécifique dans le fonctionnement global du réseau Ethereum. La couche réseau (*Network Layer*) correspond au réseau P2P où les différents nœuds se communiquent directement. C'est dans la couche donnée (*Data Layer*) que sont sauvegardées les données issues des transactions. Il s'agit des signatures numériques qui garantissent l'authenticité et l'intégrité des transactions via la cryptographie asymétrique, les empreintes numériques utilisées pour identifier les transactions et les blocs, la structure arborescente des transactions stockées dans les blocs, et les valeurs des transferts. La couche consensus (*Consensus Layer*) met en œuvre les différents mécanismes de validation des transactions, et assure la mise à jour de l'état de la blockchain. Elle est aussi responsable de la sécurité. Quant à la couche applicative (*Application Layer*), elle permet aux utilisateurs d'interagir à souhait avec la blockchain. C'est là que sont stockés et exécutés les contrats intelligents et les dApps.

Pour mieux appréhender Ethereum, nous définissons le rôle des différents éléments ou terminologies indispensables à son fonctionnement, comme suit :

a. Nœuds (comme fondation du réseau)

Selon [35], un nœud Ethereum est un ordinateur (ou machine réelle) sur lequel sont exécutés nécessairement deux (02) logiciels distincts appelés clients (client d'exécution et client de consensus), en vue de valider les blocs et les données de transactions. Le réseau Ethereum est donc

simplement le regroupement de tous ses nœuds qui se communiquent directement. Le client d'exécution capture les nouvelles transactions publiées sur le réseau, les exécute dans l'EVM, et contient la dernière base de données et l'état de toutes les données Ethereum à jour ; tandis que le client de consensus implémente le PoS et la sécurisation du réseau. La **figure 8** ci-dessous représente un nœud de moteur d'exécution couplé au client de consensus dans un réseau Ethereum.

Ainsi, dès qu'une transaction est demandée depuis un nœud, elle est publiée auprès de tous les autres nœuds du réseau qui la vérifient avant que le processus de validation ne s'enclenche. A la fin d'une validation, chaque nœud sauvegarde et met à jour la copie de chaîne de blocs.

Il existe trois (03) types de nœuds Ethereum à savoir le nœud complet, le nœud d'archive et le nœud léger.

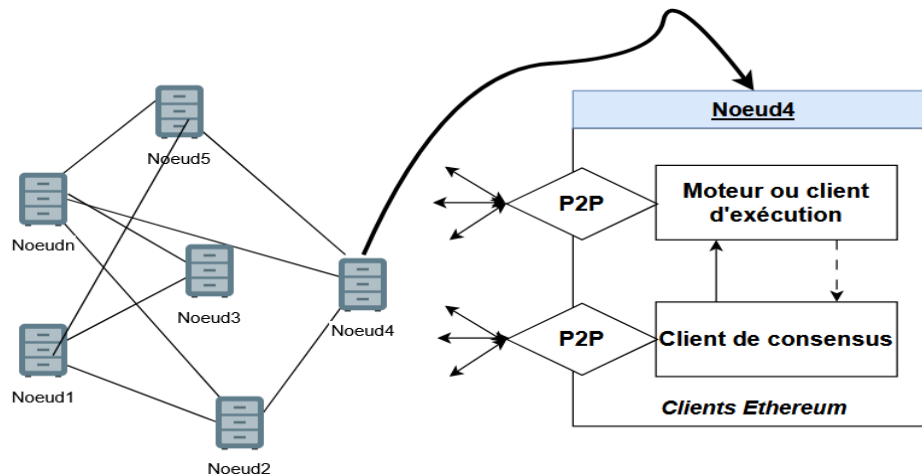


Figure 8 : Schéma simplifié d'un nœud de moteur d'exécution couplé au client de consensus dans un réseau Ethereum

b. EVM (comme environnement d'exécution)

[36] L'EVM (Ethereum Virtual Machine) est l'ordinateur virtuel unique et décentralisé sur lequel sont exécutés (sans demande de permission) les contrats intelligents ou toutes autres applications décentralisées ou (dApps). Il exécute ces codes de manière cohérente et sécurisée sur tous les nœuds Ethereum. Autrement dit, l'exécution d'une quelconque transaction ou d'un code spécifique modifie l'état de l'EVM. Et à chaque modification de l'état de l'EVM, tous les nœuds (à l'écoute

permanent) du réseau Ethereum approuvent le nouvel état et gardent une copie de celui-ci. Cela permet de garantir la transparence, la décentralisation et l'immuabilité de la blockchain.

En outre, l'EVM dispose d'une fonction de transition d'état formellement décrite dans [37]. Il implémente également plusieurs opérations spécifiques à la blockchain telles que address, balance, blockhash, etc.

c. Comptes (acteurs de base pour les transactions)

Les comptes Ethereum sont des entités disposant d'un solde en ETH. Les comptes et leur solde font partie de l'état global de l'EVM. Il existe deux (02) types de comptes Ethereum : le **compte de propriété externe (EOA- Externally Owned Account)** qui est contrôlé par toute personne ayant les clés privées et le **compte de contrat (CA- Contract Account)** qui est contrôlé par le code d'un contrat intelligent déployé sur le réseau [38]. En effet, ces types de comptes ont des caractéristiques différentes.

En effet, [38] un EOA est composé d'une paire de clés cryptographiques publique/privée qui permettent de contrôler les activités du compte. Ce compte interagit avec la blockchain en créant et en signant les transactions à l'aide de sa clé privée. La création de ce type de compte est gratuite et il offre la possibilité d'initier des transactions. Mais les transactions entre des comptes externes ne peuvent être que des transferts, réceptions, et détentions de valeurs en ETH et en jetons.

Quant au compte de contrat, sa création se fait moyennant un coût dû à l'utilisation de stockage du réseau. Ce type de compte n'a pas de clé privée et on ne peut qu'envoyer des transactions en réponse à la réception d'une transaction [38] provenant d'un EOA ou d'un autre CA. C'est-à-dire que les CA ne peuvent pas initier de nouvelles transactions par eux-mêmes. Mais, il est possible d'effectuer des transactions depuis un EOA vers un CA. Ces transactions peuvent déclencher un code pouvant exécuter plein d'actions différentes, comme transférer des jetons ou même créer un nouveau contrat.

Outre ces principales différences, l'état de tout type de compte Ethereum est structuré en quatre (04) champs illustrés par la **figure 9** ci-dessous.

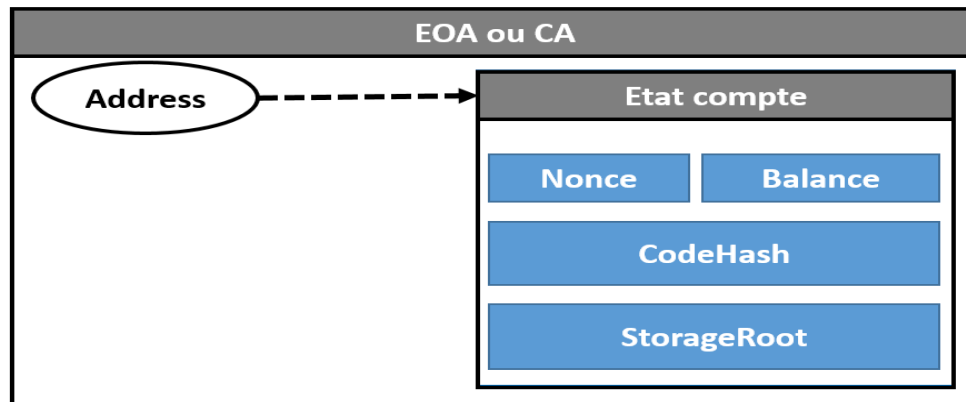


Figure 9 : Etat simplifié d'un compte Ethereum

Un compte Ethereum est identifié par une adresse unique (**Address**) qui est un code haché sur 42 caractères hexadécimaux ayant comme préfix 0x. On peut avoir comme exemple d'adresse *Account* = 0x5e97870f263700f46aa00d968721199b9bc5a129. Le champ **Nonce** représente d'une part le nombre de transactions envoyées à partir de l'adresse du compte s'il s'agit d'un EOA et d'autre part le nombre de contrats créés par le compte si c'est un CA. **Balance** est le nombre d'ETH exprimé en Wei (ou le solde actuel) possédé par cette adresse. Le **CodeHash** est l'empreinte numérique (hash) du code du compte (EOA ou CA) dans l'EVM. Contrairement aux autres champs du compte, le CodeHash n'est pas modifiable. Pour les EOA, ce champ CodeHash contient le hachage d'une chaîne vide. Mais pour le cas des CA, le code source du contrat intelligent sont hachés et stockés dans le CodeHash. Dans Ethereum, le contenu du compte est structuré sous forme d'arbre de Merkle (expliqué dans 2.3 précédemment) dont la racine est hachée sur 256 bits. Le champ **StorageRoot** (ou hachage de stockage) représente l'encodage de cet arbre. Il est vide par défaut.

d. Transactions (interactions entre comptes)

Une transaction ou demande de transaction (terme officiel) est une action initiée par un EOA (compte géré par un humain et non par un contrat). Cette action est une suite d'instructions signées cryptographiquement et dont l'exécution peut modifier l'état de l'EVM. Et pour qu'une demande de transaction modifie l'état convenu de l'EVM, elle doit être validée, exécutée et diffusée sur le réseau par un autre nœud. [39] Cette demande peut être l'envoi d'ETH depuis un compte A vers un autre compte B, la publication d'un code de contrat intelligent sur l'EVM, l'exécution du code d'un contrat intelligent à une adresse spécifique dans l'EVM, etc.

Toute transaction contient les informations suivantes :

- **from** : c'est l'adresse (émettrice) de l'expéditeur qui signe la transaction.
- **to** : c'est l'adresse du destinataire de la transaction. Dans le cas des transactions de création de contrat, l'adresse du compte du contrat n'existe pas encore et une valeur vide est donc utilisée.
- **signature** : elle est générée lorsque la clé privée de l'expéditeur signe la transaction, et confirme que l'expéditeur autorise ladite transaction.
- **nonce** : c'est le numéro de transactions dans la liste des transactions émises par l'expéditeur. On dira aussi que c'est le nombre de transactions envoyées par l'expéditeur [15].
- **value** : c'est le montant de l'ETH (en Wei) à transférer de l'expéditeur au destinataire. Dans le cas des transactions de création de contrat intelligent, cette valeur correspond au solde (Balance) de départ dans le compte de contrat nouvellement créé.
- **input data** : est un champ facultatif qui est souvent utilisé pour inclure des données arbitraires (par exemple, les nom, prénoms d'un étudiant dans le cas d'un contrat intelligent servant à enregistrer un diplôme universitaire).
- **gasLimit** : c'est la quantité maximale de gaz (estimée par l'EVM) pouvant être consommée pour l'exécution complète de la transaction. Ici, le gaz représente le coût informatique ou l'unité de mesure de la quantité d'efforts de calculs requis pour exécuter une opération sur le réseau Ethereum [40]. Une transaction simple de transfert requière 21 000 unités de gaz.
- **maxPriorityFeePerGas** (ou pourboire) : c'est la quantité maximale de gaz à inclure comme pourboire pour le validateur (uniquement) de la transaction. Ce montant incite le validateur à traiter la transaction plus rapidement.
- **maxFeePerGas** : c'est le montant maximum que l'expéditeur est prêt à payer par unité de gaz pour la transaction. Donc $maxFeePerGas = baseFeePerGas + maxPriorityFeePerGas$; où $baseFeePerGas$ = frais de base du réseau qui est ajusté dynamiquement en fonction de la congestion du réseau. Mais si à la fin de l'opération, le montant total réel payé ne vaut pas le $maxFeePerGas$ initial, le reliquat ($maxFeePerGas - (baseFeePerGas + maxPriorityFeePerGas)$) est retourné à l'expéditeur.

e. Blocs (comme conteneurs des transactions)

Un bloc est un regroupement (généralement des dizaines à des centaines) de transactions. Sur Ethereum, les blocs sont créés et engagés toutes les 12 secondes. Un bloc Ethereum a deux (02) parties : l'entête et le corps.

[41] L'entête d'un bloc contient les informations telles que slot (créneau auquel appartient le bloc), proposer_index (ID du validateur proposant le bloc), parent_root (hachage du bloc précédent), state_root (hachage racine de l'objet état), et body (qui contient aussi plusieurs autres champs). Le corps du bloc contient aussi plusieurs champs d'information tels que randao_reveal, eth1_data, attestations, voluntary_exits, execution_payload, etc.

Chaque bloc a une taille cible dont la limite est de 15 millions de gaz. Mais cette limite peut être ajustée à la hausse ou à la baisse par un facteur de 1/1024 par rapport à la limite de gaz du bloc précédent [41]. En tout état de cause, la quantité de gaz dépensée par toutes les transactions d'un bloc doit être inférieure à la limite de gaz dudit bloc.

f. ETH (comme monnaie pour payer et interagir)

[42] C'est la crypto-monnaie native d'Ethereum. L'ETH est créé par le protocole Ethereum, et non pas par un utilisateur. Chaque utilisateur d'Ethereum doit disposer d'une quantité suffisante d'ETH. Pour chaque transaction, une certaine quantité d'ETH (frais de gaz) est obligatoirement fournie par l'initiateur de ladite transaction. Cette quantité d'ETH est utilisée d'une part comme ressources pour effectuer les calculs afférant à la transaction et d'autre part comme prime pour simultanément récompenser les nœuds validateurs et empêcher les participants d'être animés de mauvaises intentions (bloquer le réseau par exemple). Étant donné que les validateurs misent (mettent en jeu) une partie de leurs ETH pour être favoris validateurs, si un d'entre eux venait à mal se comporter, ses ETH sont probablement détruits par le réseau.

L'Ether dispose de plusieurs unités de valeur. Les plus importants sont le « Wei » qui est la plus petite quantité possible d'Ether et le « Gwei » (Giga-Wei) qui est couramment utilisé pour décrire les frais de gaz lisibles par l'humain. $1 \text{ ETH} = 10^9 \text{ Gwei} = 10^{18} \text{ Wei}$.

g. Contrats intelligents

Bien que cet élément ait été présenté précédemment dans le point [2.5](#) du présent mémoire, rappelons tout de même que n'importe quel développeur peut créer un contrat intelligent, le stocker sur la blockchain et le rendre public sur le réseau, moyennant des frais (ETH) payés au réseau.

Aussi, tout utilisateur peut invoquer et exécuter ledit contrat intelligent, encore moyennant frais (ETH) payés au réseau.

Selon [43], les contrats intelligents sont assimilables à des API ouvertes pour le fait qu'un contrat intelligent peut appeler d'autres contrats et que certains d'entre eux, peuvent même déployer d'autres contrats. Par défaut, la taille maximale d'un contrat intelligent est de 24 Ko.

Les contrats intelligents sont très souvent confondus aux dApps. Cependant, pour être plus précis, une application décentralisée (dApp) est une application dont le code backend s'exécute sur un réseau décentralisé P2P, contrairement à une application traditionnelle, dont le code du backend est exécuté sur des serveurs centralisés. Une dApp combine un contrat intelligent (code backend) et une interface utilisateur (code frontend implémenté dans n'importe quel langage de programmation). C'est donc une application plus complexe des contrats intelligents [44].

En somme, nous avons proposé dans ce chapitre, un historique et des définitions de la technologie blockchain. Nous y avons également passé en revue les types de blockchain, la structuration, le fonctionnement et les mécanismes de consensus de la blockchain. Dans cette revue, nous avons découvert que les contrats intelligents ont révolutionné la manière d'effectuer les transactions blockchains, surtout dans les blockchains publiques telles qu'Ethereum qui est assez populaire.

Nous faisons dans le chapitre suivant, un état des connaissances sur les méthodes d'authentification de documents et les travaux existants en matière d'authentification de documents à l'aide de la blockchain.

CHAPITRE 3 :

**ÉTAT DE L'ART SUR
L'AUTHENTIFICATION DES
DOCUMENTS À L'AIDE DE LA
BLOCKCHAIN**

CHAPITRE 3 : ÉTAT DE L'ART SUR L'AUTHENTIFICATION DES DOCUMENTS À L'AIDE DE LA BLOCKCHAIN

Ce chapitre du présent mémoire a pour but de présenter la technologie blockchain tout en rappelant son historique. Nous y abordons la classification de la blockchain, sa structure ainsi que son fonctionnement. Aussi, on y retrouve particulièrement une présentation des travaux existants sur la sécurisation et l'authentification des documents avec la blockchain.

3.1 Authentification de documents

Définir ce que c'est.

3.2 Méthodes d'authentification de documents

Donner les méthodes ou techniques existante en matière d'authentification de documents.

Faire une comparaison (étude comparative) de ces techniques.

Pouvoir conclure que parmi tout ceci, c'est la blockchain qui est la mieux indiquer.

3.3 Travaux existants sur l'authentification de documents à l'aide de la blockchain

Il existe plusieurs techniques d'utilisation de la blockchain dans les processus de sécurisation et d'authentification de documents numériques. Dans cette partie du présent mémoire, nous présentons quelques travaux existants y relatifs.

Cas 1 : Cadre d'authentification des documents électroniques à l'aide de la technologie Blockchain dans le système gouvernemental (Isyak Meirobie et al.)

Titre d'origine : « *Framework Authentication e-document using Blockchain Technology on the Government system* ».

Dans [45], Isyak Meirobie et al. ont présenté le résultat de leurs recherches qui ont conduit à la mise en place d'une plateforme d'authentification des documents électroniques à l'aide de la technologie Blockchain dans le système gouvernemental d'Indonésie. Les problèmes ayant suscités ces recherches sont le manque de sécurité dans le stockage de toutes les données des documents, les redondances profondes de données et la présence de tierces parties qui interfèrent dans les transmissions de documents. Afin de minimiser la falsification des documents et de maximiser les documents électroniques du gouvernement d'une manière moderne et sécurisée, la méthode a été de combiner la blockchain, des smart contracts (contrats intelligents) et des Decentralized Autonomous Organization (DAO ou type de plus complexe des smart contracts).

En effet, les gouvernements pourraient charger un ensemble de données et de documents sur une blockchain publique (sans nécessiter d'autorisation) et utiliser des signatures pour signer les transactions. Les signatures sont librement accessibles via le site web de l'institut. Et chaque gouvernement qui souhaite confirmer l'authenticité d'un document via la blockchain peut s'en assurer grâce à sa transcription numérique, tout en vérifiant que la transaction qui l'intègre à la blockchain est signée par le gouvernement lui-même. Au lieu de stocker toutes les données complètes sur la blockchain, seul le hachage de la signature SHA256 des données est stocké. Cela élimine la nécessité d'un stockage massif tout en garantissant l'intégrité et la vérification de toutes les données. La blockchain publique utilisée est sans licence, basée le processus de consensus PoA.

Concrètement, les auteurs de cette étude ont mis en place une interface utilisateur simple dénommé Go-Chain (Government Blockchain). Go-Chain est construit en 3 couches essentielles à savoir la couche de vérification, la couche des services de logique métier, et la couche d'accès/persistance de données. En amont, les documents (pdf ou word) gouvernementaux peuvent être téléversés, signés (via clé privée), transcrits numériquement en json et stockés (après calcul de la racine de Merkle) sur la blockchain. La transcription sous forme json peut être distribué au public. En retour, le public peut présenter le document haché à toute entreprise ou institution comme preuve valable. Mais, pour tout de même vérifier la validité ou l'authenticité d'un document via la blockchain, le public peut téléverser le document numérique gouvernemental dans Go-Chain, en y saisissant une clé privée. Après recalcule de la racine de Merkle, le cadre compare cette racine recalculée avec la racine de Merkle auparavant stockée sur la blockchain et signale si elle a été signée par une institution légitime. Pour la signature, l'auteur a utilisé un Digital Signature Algorithm (DSA) avec une courbe P-256. Et lorsque le document chargé par le public est valide, la clé publique, l'empreinte digitale SHA265 et d'autres données apparaissent sur l'écran de vérification Go-Chain.

En termes d'outils et de technologies, les auteurs ont utilisé HTML5, CSS3, JavaScript (ES6), Python 3, le microframework Flask et des serveurs HTTP.

Cas 2 : La Blockchain pour la Sécurisation des E-livrets scolaires (Ana BAKHOUM)

Ana BAKHOUM [15] a proposé, au profit du système d'enseignement moyen et secondaire du Sénégal, la dématérialisation du livret scolaire (d'où le E-livret). Le livret scolaire est un document administratif au format papier qui permet de répertorier les notes des élèves de la classe de sixième

à la classe de terminal. Le même livret scolaire est transféré dans chaque établissement d'enseignements fréquenté par l'élève. Cette dématérialisation a constitué à la mise en place d'un système de recueil et de stockage (dans une base de données relationnelle MySQL hébergée par un serveur) des informations qui étaient dans le livret en papier. Dans cette dynamique, la problématique majeure traitée par l'auteur est comment assurer la fiabilité, l'authenticité, la transparence et la sécurité des E-livrets ? quelle architecture idéale, quel type de stockage utilisé ?

Dans ses travaux en lien avec cette problématique, l'auteur a fait une revue de littérature sur la technologie blockchain en générale et la blockchain Ethereum en particulier. Il a aussi passé en revue, la question de la sécurité informatique et celle notamment appliquée à la technologie blockchain. De ce qui est de la sécurité informatique en générale, il s'agit des obligations d'authentification, d'intégrité, de confidentialité, de disponibilité et de non-répudiation. Celle-ci pourrait faire face partiellement aux vulnérabilités, menaces, risques et attaques dans la blockchain. Car dans la pratique, il existe de multiples attaques qui manipulent directement ou indirectement le mécanisme de récompense (des mineurs), donnant ainsi d'injustes avantages aux mineurs de plus grandes tailles aux détriment des petits mineurs.

Selon les standards de la norme ISO/TC 307 [46], plusieurs propriétés de sécurité sont intégrées dans la blockchain, notamment dans les applications basées sur les Distributed Ledger Technologies (DLT). Ce sont entre autres les propriétés :

- d'intégrité qui assure la protection de données contre toute modification après création ;
- d'authenticité qui permet de vérifier, qui enregistre une transaction dans le registre ;
- de confidentialité qui garantit que le registre est uniquement consultable par ceux qui y sont autorisés ;
- de disponibilité qui permet d'assurer la disponibilité à tout moment de toute transaction déjà enregistrée ;
- d'ordonnancement des événements rendant impossible le changement d'ordre des enregistrements dans le registre avec l'horodatage ;
- de « trusted-server less » permettant à la blockchain de toujours fonctionner malgré l'absence de serveur de confiance ;
- etc.

Dans la même logique, des mécanismes de sécurité ont été intégrés dans la blockchain tels que la cryptographie (surtout asymétrique), la signature numérique, le hachage.

L'étude a, à terme, permis de mettre en place un système décentralisé de sécurisation des E-livrets scolaires (SDSEL) en s'appuyant sur la technologie Blockchain, particulièrement sur Ethereum. Outre leur sécurisation, le SDSEL permet de valider ou de vérifier les E-livrets scolaires des élèves. La démarche a été de :

- développer le système (déjà existant et utilisé dans l'étude) de gestion des livrets électroniques (SGLE) qui permet de saisir et traiter, des informations des livrets scolaires à savoir les établissements, les élèves, les notes, les appréciations du conseil, etc.
- développer l'application décentralisée (dApp) dénommée « SDSEL » pour la sauvegarde des E-livrets dans la blockchain ;
- importer les informations des E-livrets depuis la base de données du SGLE vers la Blockchain des E-livrets ;
- consulter les listes et statistiques des élèves et leur livret ; cela permet de vérifier la conformité avec le livret généré par le SGLE ;
- développer (en perspective de l'étude) les contrats qui permettront à l'office du bac de générer automatiquement la liste des élèves inscrits en terminale qui servira à l'organisation de l'examen de baccalauréat ;
- déployer (en perspective de l'étude) le SDSEL dans la blockchain publique Ethereum afin qu'elle soit accessible par tous les établissements d'enseignements.

En termes d'outils et de technologies pour la mise en place du Dapp SDSEL, l'auteur a utilisé l'API JavaScript Web3, l'API JSON RPC, le langage de programmation Solidity, l'IDE Remix-IDE, le framework Truffle Framework qui intègre GANACHE, les frameworks Angular et Spring.

En s'appuyant sur ces études précédentes et en tenant compte des différents contextes, voyons si la blockchain est-elle plus indiquée pour résoudre la problématique de sécurisation et d'authentification de documents administratifs.

3.4 Discussion

Après analyse de la synthèse des travaux réalisés par Ana BAKHOUM et les co-auteurs Isyak Meirobie et al., nous décelons quelques points communs.

En effet, ces auteurs ont utilisé une **blockchain publique** dans le cadre de leurs travaux. Par ailleurs, si *Isyak Meirobie et al.* n'ont pas, dans leur synthèse, mentionné explicitement laquelle des blockchains publiques qu'ils ont expérimenté, *Ana BAKHOUM* a, quant à elle, utilisé Ethereum.

Aussi, les deux (02) acteurs ont utilisé la notion de **contrat intelligent**. Des contrats intelligents ont été développés afin d'automatiser des traitements sur la blockchain. Il s'agit par exemple de :

- la transcription numérique de documents, le hachage, la signature numérique et le stockage des données dans la blockchain ;
- l'importation et le stockage des E-livrets dans la blockchain ;
- la vérification de l'authenticité des E-livrets et documents gouvernementaux depuis la blockchain ;
- etc.

Cela dénote de l'utilisation des propriétés de stockage décentralisé et de transparence (fiabilité) et d'intégrité de la technologie blockchain.

De plus, [15] et [45] ont travaillé sur des **documents électroniques**, même s'il s'agit de différents types de documents. Ramenant à notre sujet du présent mémoire, cela pourrait être intéressant dans la mesure où l'Administration utilise couramment des documents électroniques ; sachant qu'un document est dit électronique, s'il est créé directement au format numérique (sans lien direct avec un support physique) ou converti (numérisé).

En outre, dans les différentes solutions mises en place, toutes les données sujettes à authentification et devant être sécurisées **ne sont pas entièrement stockées** sur la blockchain, évitant ainsi les stockages massifs.

Hormis ces points communs, quelques applications suscitent des réflexions d'ordre technique et juridique. En effet, comment opérer le choix du processus de consensus qui puisse cadrer avec le type de blockchain adopté ([45] a utilisé le PoA) ? relativement au contexte de notre présent projet, le cadre juridique national permet-il d'exploiter des signatures numériques et/ou électroniques de documents administratifs ? stocker des informations issues de documents administratifs dans la blockchain (même hachées et/ou cryptées) ne met-il pas en cause la souveraineté de l'Etat ? afin d'assurer la sécurité et l'intégrité du réseau de blockchain, il y a des coûts connexes liés entre autres aux minages des transactions blockchain. **En adoptant une**

blockchain publique pour solutionner la sécurisation et l'authentification de documents administratifs, l'Administration supportera-t-elle ces coûts de façon pérenne ?

En tout état de cause, l'adoption de la blockchain nous semble appropriée pour la résolution de notre problématique. Dans le chapitre suivant, nous présentons, en tenant compte de ce qui précède, notre approche qui permet d'authentifier un type spécifique de document administratif.

CHAPITRE 4 :

**APPROCHE D'AUTHENTIFICATION DE
DOCUMENTS À L'AIDE DE LA
BLOCKCHAIN**

CHAPITRE 4 : APPROCHE D'AUTHENTIFICATION DE DOCUMENTS À L'AIDE DE LA BLOCKCHAIN

Je liste les étapes (peut-être en sous points) pour la réalisation. Là je fixe le type de document administratif sur lequel nous travaillons.

CHAPITRE 5 :

IMPLÉMENTATION DE L'APPROCHE

CHAPITRE 5 : IMPLÉMENTATION DE L'APPROCHE

5.1 Protocole d'implémentation

Décrire l'environnement de dev, les outils utilisés pour l'implémentation, les éléments de conceptions.

5.2 Présentation de la solution

Je présente des parties de la solution implémentée.

5.3 Discussion des résultats

Dire si oui, non ou partiellement la solution proposée répond à la problématique. Est-ce que la solution répond aux hypothèses posées plus haut ?

CONCLUSION ET PERSPECTIVES

CONCLUSION ET PERSPECTIVES

RÉFÉRENCES

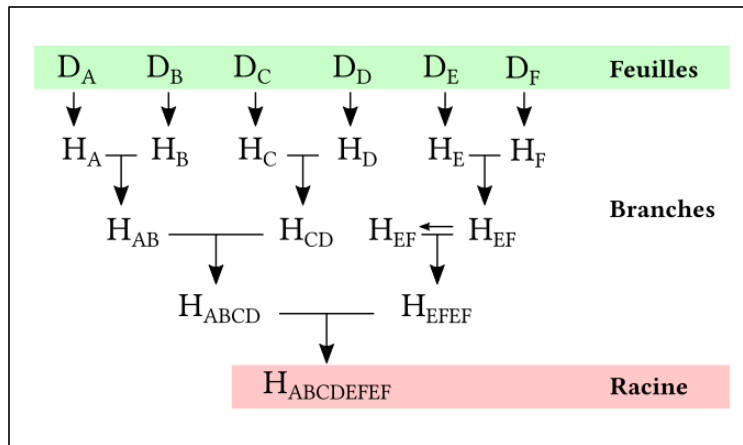
- [1] Ministère de l'Economie et des Finances du Burkina Faso, « ALERTE VIGILANCE ». Consulté le: 26 janvier 2025. [En ligne]. Disponible sur: <https://www.facebook.com/share/p/1BP46UYXF9/>
- [2] DCRP/MFPTPS, « Ministère de la Fonction Publique, du Travail et de la Protection Sociale ». Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://www.facebook.com/share/p/KHPyDxy6A1zMniK1/?mibextid=oFDknk>
- [3] Douanes du Burkina Faso, « Service de Communication et des Relations Publiques de la Direction Générale des Douanes ». Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://www.facebook.com/share/p/15f4cKVZ3P/>
- [4] DCRP/MAECR-BE, « Ministère des Affaires Etrangères du Burkina Faso ». Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://www.facebook.com/share/p/yqFV712VwsjcHxne/?mibextid=oFDknk>
- [5] Direction générale de l'INSD, « Démenti de recrutement d'étudiants ». Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://www.facebook.com/share/p/DYayYQThHP2NLDbF/?mibextid=oFDknk>
- [6] B. Bertrand, « Loi 051 portant sur l'accès à l'information publique - CSC - BURKINA FASO ». Consulté le: 16 septembre 2024. [En ligne]. Disponible sur: <https://www.csc.bf/index.php/textes-de-reference/lois/item/76-loi-051-portant-sur-l-acces-a-l-information-publique>
- [7] A. Back *et al.*, « Enabling Blockchain Innovations with Pegged Sidechains », p. 25, oct. 2014.
- [8] S. Haber et W. S. Stornetta, « How to time-stamp a digital document », *J. Cryptol.*, vol. 3, n° 2, p. 99-111, janv. 1991, doi: 10.1007/BF00196791.
- [9] D. Bayer, S. Haber, et W. S. Stornetta, « Improving the Efficiency and Reliability of Digital Time-Stamping », in *Sequences II*, R. Capocelli, A. De Santis, et U. Vaccaro, Éd., New York, NY: Springer, 1993, p. 329-334. doi: 10.1007/978-1-4613-9323-8_24.
- [10] « La première blockchain de l'histoire date de 1995, et elle est imprimée sur papier », Le Monde.fr. Consulté le: 17 septembre 2024. [En ligne]. Disponible sur: https://www.lemonde.fr/big-browser/article/2018/09/01/la-premiere-blockchain-de-l-histoire-date-de-1995-et-elle-est-imprimee-sur-papier_5349082_4832693.html
- [11] R. Subramanian et T. Chino, « The State of Cryptocurrencies, Their Issues and Policy Interactions », *J. Int. Technol. Inf. Manag.*, vol. 24, n° 3, p. 40, janv. 2015, doi: 10.58729/1941-6679.1045.
- [12] S. Nakamoto, « Bitcoin: A Peer-to-Peer Electronic Cash System », p. 9.
- [13] Ferr. Godebargue et R. Rossat, « Principes clés d'une application Blockchain », *EM Lyon Bus. Sch.*, p. 50, déc. 2016.
- [14] « Blockchain », *Wikipédia*. 11 janvier 2025. Consulté le: 19 septembre 2024. [En ligne]. Disponible sur: <https://fr.wikipedia.org/w/index.php?title=Blockchain&oldid=222001944#Histoire>
- [15] A. Bakhoun, « La Blockchain pour la Sécurisation des E-livrets scolaires », p. 103, 2019.
- [16] I. Bashir, *Mastering Blockchain*. Packt Publishing Ltd, 2017.
- [17] J.-L. Jonnaert, « Quelles différences entre blockchain publique et blockchain privée ? », Cryptoast. Consulté le: 29 septembre 2024. [En ligne]. Disponible sur: <https://cryptoast.fr/differences-blockchain-publique-blockchain-privee/>
- [18] H. Anwar, « Blockchain Consortium: Top 20 Consortia You Should Check Out », 101 Blockchains. Consulté le: 2 octobre 2024. [En ligne]. Disponible sur: <https://101blockchains.com/blockchain-consortium/>
- [19] « Qu'est-ce qu'un arbre Merkle? », Bit2Me Academy. Consulté le: 18 décembre 2024. [En ligne]. Disponible sur: <https://academy.bit2me.com/fr/qu%27est-ce-qu%27un-arbre-merkle/>
- [20] A. A. Oussama, « CHAPITRE III : État de l'art de la Blockchain », ResearchGate. Consulté le: 21 octobre 2024. [En ligne]. Disponible sur: https://www.researchgate.net/publication/335174496_CHAPITRE_III_Etat_de_l'art_de_la_Blockchain

- [21] « Éléments Fondamentaux d'un Bloc dans la Blockchain - W3r.one Magazine ». Consulté le: 21 octobre 2024. [En ligne]. Disponible sur: <https://w3r.one/fr/blog/blockchain-web3/architecture-blockchain/conception-de-blocs/elements-fondamentaux-bloc-blockchain>
- [22] B. Anders, « Blockchain Demo ». Consulté le: 23 janvier 2025. [En ligne]. Disponible sur: <https://andersbrownworth.com/blockchain/hash>
- [23] É. Larousse, « Définitions : consensus - Dictionnaire de français Larousse ». Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://www.larousse.fr/dictionnaires/francais/consensus/18357>
- [24] « Consensus : Définition simple et facile du dictionnaire ». Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://www.linternaute.fr/dictionnaire/fr/definition/consensus/>
- [25] « Qu'est-ce que le consensus ? Guide du débutant », Qu'est-ce que le consensus ? Guide du débutant. Consulté le: 15 septembre 2024. [En ligne]. Disponible sur: <https://crypto.com/fr/university/consensus-mechanisms-explained>
- [26] « How to Agree: Different Types of Consensus for Blockchain », How to Agree: Different Types of Consensus for Blockchain. Consulté le: 1 février 2025. [En ligne]. Disponible sur: <https://crypto.com/en/university/different-types-of-consensus-for-blockchain>
- [27] « Qu'est-ce qu'une attaque à 51% et quels sont les risques ? », coinbase. Consulté le: 3 février 2025. [En ligne]. Disponible sur: <https://www.coinbase.com/fr/learn/crypto-glossary/what-is-a-51-percent-attack-and-what-are-the-risks>
- [28] « Blockchain : qu'est-ce qu'un Smart Contract et à quoi ça sert ? », LeMagIT. Consulté le: 5 février 2025. [En ligne]. Disponible sur: <https://www.lemagit.fr/conseil/Blockchain-quest-ce-quun-Smart-Contract-et-a-quoi-ca-sert>
- [29] « Que sont les contrats intelligents ? | Les contrats intelligents expliqués | Kraken ». Consulté le: 8 février 2025. [En ligne]. Disponible sur: <https://www.kraken.com/fr/learn/what-are-smart-contracts>
- [30] « Que sont les contrats intelligents sur la blockchain ? | IBM ». Consulté le: 7 février 2025. [En ligne]. Disponible sur: <https://www.ibm.com/fr-fr/topics/smart-contracts>
- [31] « Smart contract : définition et fonctionnement ». Consulté le: 7 février 2025. [En ligne]. Disponible sur: <https://www.captaincontrat.com/contrats-commerciaux-cgv/contrats-commerciaux/smart-contract-definition-et-fonctionnement-me-beaubourg-avocats>
- [32] « Ethereum », *Wikipédia*. 22 janvier 2025. Consulté le: 10 février 2025. [En ligne]. Disponible sur: <https://fr.wikipedia.org/wiki/Ethereum>
- [33] T. Bourbotte, « C'est quoi Ethereum ? Nos explications pour tout savoir sur cette blockchain et sa cryptomonnaie ETH », Cryptoast. Consulté le: 10 février 2025. [En ligne]. Disponible sur: <https://cryptoast.fr/fiche-ethereum/>
- [34] S. S. Kushwaha, S. Joshi, D. Singh, M. Kaur, et H.-N. Lee, « Systematic Review of Security Vulnerabilities in Ethereum Blockchain Smart Contract », *IEEE Access*, vol. 10, p. 6605-6621, 2022, doi: 10.1109/ACCESS.2021.3140091.
- [35] « Nœuds et clients », ethereum.org. Consulté le: 12 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/nodes-and-clients/>
- [36] « Machine virtuelle Ethereum (EVM) », ethereum.org. Consulté le: 12 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/evm/>
- [37] T. Takenobu, « Ethereum EVM illustrated », p. 116.
- [38] « Comptes Ethereum », ethereum.org. Consulté le: 12 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/accounts/>
- [39] « Transactions », ethereum.org. Consulté le: 15 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/transactions/>
- [40] « Gaz et frais », ethereum.org. Consulté le: 15 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/gas/>
- [41] « Blocs », ethereum.org. Consulté le: 17 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/blocks/>

- [42] « Introduction à l'ether », ethereum.org. Consulté le: 12 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/intro-to-ether/>
- [43] « Introduction aux contrats intelligents », ethereum.org. Consulté le: 12 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/smart-contracts/>
- [44] « Introduction aux dApps », ethereum.org. Consulté le: 12 février 2025. [En ligne]. Disponible sur: <https://ethereum.org/fr/developers/docs/dapps/>
- [45] I. Meirobie, A. P. Irawan, H. T. Sukmana, D. P. Lazirkha, et N. P. L. Santoso, « Framework Authentication e-document using Blockchain Technology on the Government system », *Int. J. Artif. Intell. Res.*, vol. 6, n° 2, Art. n° 2, déc. 2022, doi: 10.29099/ijair.v6i2.294.
- [46] « ISO/TC 307 - Blockchain and distributed ledger technologies ». Consulté le: 31 octobre 2024. [En ligne]. Disponible sur: <https://www.iso.org/committee/6266604/x/catalogue/p/1/u/1/w/0/d/0>

ANNEXES

Annexe 1 : exemple d'arbre de Merkle dans Bitcoin



Source : <https://cryptoast.fr/wp-content/uploads/2020/08/merkle-tree-general.png>

Annexe 2 : circuit d'une transaction

