

BLOCKCHAIN

SOMMAIRE

I. INTRODUCTION.....	2
II. HASH.....	2
III. Algorithme de consensus.....	3
IV. Proof of Work.....	3
V. Bitcoin.....	3
VI. SMART CONTRACTS.....	5

I. INTRODUCTION

La Blockchain est une technologie de stockage et de partage de données transparente et sécurisée, fonctionnant sans organe central de contrôle. C'est une sorte de base de données distribuée qui contient l'historique de tous les échanges effectués entre ses utilisateurs depuis sa création. La blockchain est comme un registre public infalsifiable.

Il existe plusieurs sortes de blockchains. Il y a les blockchains publiques (ouvertes à tous), privées (dont l'accès limité à un certain nombre d'acteurs) ou encore consortium (un mélange des deux précédentes).

Voici trois exemples de blockchains et leurs différences:

- Bitcoin (BTC): transfert de bitcoin;
- Ethereum (ETH): transfert d'argent avec plus d'informations que le BTC;
- HyperLedger Fabric: utilisé par les entreprises.

Une base de données est plus rapide que les blockchains.

II. HASH

1. Définition

Un hash est une empreinte numérique générée à partir d'une donnée. C'est une suite fixe de caractères, obtenue en appliquant une fonction mathématique à une donnée d'entrée.

2. Propriétés

Un hash est unidirectionnel. En effet, il est impossible de retrouver les données initiales à partir du hash. Sa longueur est fixe et il est sensible aux modifications.

Exemple: Fonction de hachage SHA-256

SHA = Secure Hash Algorithm et 256 car la sortie est une chaîne de 256 bits (64 caractères max).

Il existe deux types de cryptage:

- Symétrique (symetric encryption): A et B ont la même clé
- Asymétrique (asymetric encryption): Le message de A est chiffré avec la clé publique de B et B le déchiffre avec sa clé privée

III. Algorithme de consensus

Un algorithme de consensus permet à tous les participants (nœuds) d'un réseau blockchain de s'accorder sur l'état du registre partagé (blockchain).

Les objectifs principaux sont la sécurité (empêcher la falsification ou le double-spending), la fiabilité (s'assurer que tous les nœuds partagent la même version de la blockchain) et la décentralisation (supprimer le besoin d'une autorité centrale).

Les deux principaux algorithmes de consensus sont le Proof of Work et le Proof of Stake.

IV. Proof of Work

Le hash puzzle: Un hash est une empreinte unique générée par une fonction mathématique, basée sur les données du bloc. Le hash doit respecter une condition spécifique, par exemple: commencer par un certain nombre de 0.

V. Bitcoin

En 2008, Satoshi Nakamoto publie le whitepaper Bitcoin et le 3 janvier 2009 est miné le premier bloc (Genesis).

Voyons quelques notions de base.

Clé privée: Suite de 256 bits (généralement représentée en hex, ex. 64 caracteres)

Clé publique: Issue de la clé privée via Elliptic Curve Digital Signature Algorithm (ECDSA)

Adresse: Hash de la clé publique

Diffusion dans le réseau: Le réseau Bitcoin est P2P (Peer to Peer). Chaque noeud relaie la transaction à ses pairs, qui la stockent dans leur mempool (si valide)

Validation initiale: Les noeuds vérifient → Pas de double dépense, signatures correctes, frais suffisants (optionnel mais incitatif)

Architecture du réseau: Pas de serveur central → chaque nœud est à la fois client du serveur.

Propagation → Les messages (transactions, blocs) se diffusent en mode inondation

Mempool: BDD locale contenant les transactions non confirmées. Les mineurs piochent dans leur mempool pour construire un bloc

Fees et priorité: Les transactions offrent des frais (satoshis/byte ou sat/vByte). Plus les frais sont élevés, plus la probabilité d'être incluse rapidement est grande.

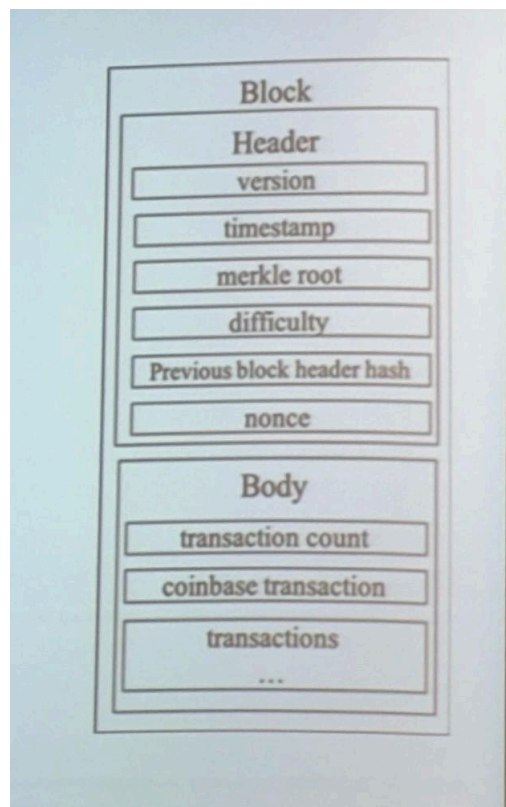
Structure d'un bloc:

Entête: Version, hash du bloc précédent, Merkle Root, Timestamp, Bits, Nonce.

Liste des transactions: Coinbase (récompense) + Transactions ordinaires (ordonnées, 2 à 3k tx en moyenne par bloc)

Chaînage: Chaque bloc inclut le hash du bloc précédent -> formation de la chaîne. Toute modification d'un bloc ancien invaliderait tous les blocs suivants

Exemples chiffrés: Taille d'un bloc : environ 1 à 2 Mo. Approx. environ 10 minutes pour miner un bloc (cible ajustée toutes les 2016 blocs)



Algorithme de consensus	Blockchain	Fonctionnement	Pros	Cons
Proof of Work	Bitcoin Litecoin	- résoudre un problème mathématique complexe - le premier crée le bloc - hachage et résultat spécifique	- sécurité +++ - décentralisation	- consommation énergétique - scalabilité faible - pas rapide
Proof of Stake	Ethereum 2.0 Solana Cardano	- les validateurs doivent verrouiller leurs crypto (miser) pour être sélectionné pour valider le bloc	- moins d'énergie - rapide	- centralisation - moins sécurisé
Proof of Authority	Vechain	- choix basé sur autorité/réputation	- rapide - moins d'énergie	- centralisation - privé - besoin de faire confiance aux nodes
PBFT	Hyperledger Fabric	- système de vote	- sécurisé	- latence - pas utilisable pour PB
Proof of Burn			- moins d'énergie	- pas rapide
Delegated Proof of Stake				
Proof of Reputation				
Tendermint				

VI. SMART CONTRACTS

Un contrat intelligent est un programme informatique qui s'exécute automatiquement lorsque les conditions prédéfinies sont remplies. Il est stocké sur la blockchain et garantit la transparence, la sécurité et l'immuabilité. Ils sont introduits dans les années 90 par Nick Szabo, mais réellement appliqués après la blockchain avec l'Ethereum. Ils remplacent les contrats traditionnels dans un contexte de blockchain. C'est un programme sans autorité, qui contrôle directement les valeurs numériques. Ils ressemblent à des instructions « si-alors » qui évaluent automatiquement des conditions prédéfinies et effectuent des transactions.

Architecture technique:

Déploiement: Le contrat est déployé sur la blockchain via une transaction

Exécution: Déclenchée par des transactions ou appels (appel de fonction)

Consensus: Les nœuds du réseau valident l'exécution via le mécanisme de consensus (Proof of Work, Proof of Stake, etc.)

Acteurs impliqués:

Développeurs: Écrivent le code du contrat intelligent.

Utilisateurs: Interagissent avec le contrat en initiant des transactions

Mineurs/Validateurs: Valident et enregistrent les transactions sur la blockchain.

Fonctionnement Technique des contrats intelligents:

Langages de programmation: Les contrats intelligents sont généralement écrits en Solidity, un langage de programmation influencé par JS, Python et C++, conçu spécifiquement pour la plateforme Ethereum.

Compilation et déploiement: Le code source est compilé en bytecode, qui est ensuite déployé sur la blockchain Ethereum. Chaque contrat déployé possède une adresse unique sur le réseau, permettant aux utilisateurs d'interagir avec lui.

Machine Virtuelle Ethereum (EVM): L'EVM est responsable de l'exécution du bytecode des contrats intelligents. Elle garantit que chaque nœud du réseau exécute le contrat de manière identique, assurant ainsi la cohérence et l'intégrité du réseau

Une fois déployée, l'instance du contrat est synchronisée sur chaque nœud du réseau. Si une transaction fait évoluer l'état du contrat, ce nouvel état sera répliqué sur chaque nœud du réseau.

Cas d'utilisation des smart contract:

Finance décentralisée: les contrats intelligents permettent la création de plateformes financières décentralisées offrant des services tels que les prêts, les emprunts et les échanges sans intermédiaire.

Assurance: Automatisation des processus de réclamation et de paiement, réduisant ainsi les délais et les coûts administratifs.

Gestion de la chaîne d'approvisionnement: suivi transparent et immuable des produits tout au long de la chaîne d'approvisionnement, garantissant l'authenticité et la qualité des produits.

Blockchain Basics

Compte-Rendu

Sommaire

I. Structure d'un bloc	2
II. Arbre de Merkle (Transactions Réelles)	3
III. Proof of Work et Minage	4
IV. Validation de la Chaîne de Blocs	5
V. Halving et Récompenses de Blocs	6
VI. Principales Attaques	7
VII. Exploration d'un Bloc en Ligne	8
VIII. Frais de Transaction et Mempool	8
IX. Recherche Complète d'un Bloc	9
X. Sécurité et Décentralisation (Noeuds)	10
XI. Hard Fork vs Soft Fork	11
XII. SegWit et la Notion de "Weight"	11
XIII. Clés Privées, Clés Publiques, Adresses et WIF	12
XIV. Introduction au Lightning Network	13
XV. Installation d'un Wallet et Transactions sur le Testnet	13

I. Structure d'un bloc

Éléments de cours

1. C'est un groupement de transactions validées. Les deux grandes parties sont le header et le body.
2. On y trouve la version, indiquant les règles de validation à suivre, le hash du bloc précédent, permettant de lier ce bloc au bloc précédent, la racine de Merkle, résumant toutes les transactions du bloc, l'horodatage, indiquant la date et l'heure de minage du bloc, la difficulté de minage du bloc et le nonce, servant de preuve de travail.
3. Chaque bloc contient le hash du bloc précédent, ils sont donc tous liés entre eux. De ce fait, toute modification sur un bloc modifiera son hash et entraînerait forcément des modifications sur les blocs suivants et demanderait de tous les recalculer, ce qui est impossible. Il y a donc une sécurisation à ce niveau-là et une falsification presque impossible.
4. Le hash du bloc et son id désignent essentiellement la même chose.
5. La limite de 1 MB avait pour objectif de trouver un équilibre entre la vitesse des transactions et la sécurité du réseau. Avec SegWit, la taille des blocs peut maintenant atteindre 1.6 MB.

Questions Pratiques

1. BLOC #14000
 - a. **Hauteur:** 14000 ;
Hash:000000002d9050318ec8112057423e30b9570b39998aacd00ca648216525fce3
Hash du bloc précédent:
0000000096b3f9b81c07ecd73d9c531071842364143e5ae4a88f6180de3d4ea0
 - b. La version est 0x1. Elle indique les règles de validation du bloc à suivre.
2.
 - a. Le hash final du bloc serait totalement modifié puisque tous les hash dépendent des autres.
 - b. Le hash ne respecterait plus la difficulté et ne commencerait plus par le nombre de zéros imposés, le rendant ainsi invalide.

II. Arbre de Merkle (Transactions Réelles)

Éléments de cours

1. C'est une structure sous forme d'arbre qui utilise des fonctions de hachage et organisant les données de manière hiérarchique. Bitcoin utilise ce concept car il permet de réduire la taille des données stockées dans la blockchain et de vérifier efficacement les transactions.
2. Plus rapide et plus économe
3. La Merkle Proof est une méthode permettant de prouver qu'un élément fait partie d'un groupe d'éléments plus large. Un nœud SPV utilise cette preuve pour vérifier qu'une transaction est dans un certain bloc sans avoir à télécharger toute la blockchain.
4. Si on modifie un seul bit, le hash de la transaction sera modifié et les changements se propageront jusqu'en haut de l'arbre et les nœuds parents seront modifiés jusqu'à la racine, qui sera complètement différente de celle de base.

Questions Pratiques

1. BLOC #700000
 - a. **H12:**22633F97F634FCA6907584B0A9959E01287B4E4E7D8FDB30A3DF36478D77791B
H34:C842044C887C44CB6E504260B1E39167F984D80AF82A7CC04F6D6EA1819CE035
 - b. 1571442CD2BC6771997DB0794D930AF04C20E884889427EEEDA1CE3BCFA9F1C7
 - c. La racine réelle du bloc est
1f8d213c864bfe9fb0098cecc3165cce407de88413741b0300d56ea0f4ec9c65.
Les deux racines diffèrent puisque dans celle que l'on a trouvée, on ne prend en compte que 4 transactions et pas toutes les transactions du bloc.
2. On calcule la racine en combinant les hachages fournis avec celui de la transaction. Si la racine calculée correspond à celle du bloc, la transaction est validée comme faisant partie du bloc.

III. Proof of Work et Minage

Éléments de cours

1. Le Proof of Work est un algorithme de consensus. Les mineurs doivent trouver un nonce respectant des conditions précises afin de valider le bloc. Les mineurs sont ensuite récompensés.
2. Permet d'équilibrer le rythme de validation des blocs, pour maintenir le ratio d'un bloc toutes les 10 minutes. Si un bloc est miné trop rapidement, la difficulté est augmentée et si trop lentement, la difficulté est diminuée.
3. Plus un mineur possède de hashrate, plus il a de chances de trouver le nonce et donc de valider les transactions d'un bloc.
4. Il n'existe pas vraiment de méthode pour trouver le nonce, si ce n'est d'essayer des valeurs au hasard jusqu'à trouver la bonne, ce qui est donc totalement aléatoire. Cela consomme beaucoup d'énergie.

Questions Pratiques

1. **En-tête de bloc:** Bloc #100, Hash précédent: 000abc..., MerkleRoot: 123xyz..., Nonce: ???
 - a. Difficulté = 3 (le hash doit commencer par 3 zéros)
 - b.

Nonce = 1: c46e2a481854f6f5fd5b80f0906dd18c591fb89833a888873529427b242e4681

Difficulté non respectée

Nonce = 2: e101d6099b17272aed22f0f10126dbc7b81b830bb6cd6dc01db3dcb9b6ff94c3

Difficulté non respectée

Nonce = 3: 99f7be2f94573dbacd99f0fcea8a2608ae16bec21950e57bcdfa24fc3e8bff94

Difficulté non respectée

2.
 - a. Il peut modifier l'horodatage (dans une certaine limite), la première transaction (récompense du mineur) ou encore l'ordre des transactions
 - b. Ces modifications sont limitées et encadrés par le protocole pour éviter la fraude et la triche ainsi

IV. Validation de la Chaîne de Blocs

Éléments de cours

1. Pour qu'un bloc soit valide, il faut que son hash respecte la difficulté imposée (le hash doit commencer par le bon nombre de zéros) et qu'il soit bien calculé en fonction des données du bloc (hash précédent, merkle root, horodatage...). Il doit contenir le hash du bloc précédent, que toutes les transactions soient valides, que le Merkle Root soit le bon (représentation de toutes les transactions), que la transaction de récompense soit correcte et que le bloc respecte la taille maximale.
2. Un bloc invalide se répercute sur tous les blocs qui suivent puisque le hash de ceux-ci est calculé en fonction du hash précédent. Ainsi, si un bloc est invalide, tous ceux qui suivent n'ont plus le bon hash.
3. D'après le consensus Proof of Work, en cas de fork temporaire, c'est la chaîne la plus longue qui est adoptée.

Questions Pratiques

1. 3 blocs fictifs avec une erreur

BLOC #1:

Hash précédent: 000,

Merkle Root:

61584028b0bc01580abd25ea389dd73649851dbfa34d8a88251e5bb81d54a21d

Transactions: Perso 1 → Perso 2: 1 BTC; Perso 3 → Perso 4: 7 BTC

Nonce: 458

Hash: 00666445c03cddb3fbf952700ed786e96ccc8b11b861c6df3f8b90fea25a087c

BLOC #2:

Hash précédent:

00666445c03cdbb3fbf952700ed786e96ccc8b11b861c6df3f8b90fea25a087c,

Merkle Root: 1e8b9c4fc8d229a0f81cbe531ce0c1e1eaf1ec5f192df94875f91166f275374d

Transactions: Perso 2 → Perso 5: 0.5 BTC; Perso 4 → Perso 1: 2 BTC

Nonce: 46

Hash: 00884bd7cacd6325687737138ee530504788914af9df5d534efa27851c01e865

BLOC #3:

Hash précédent:

00666445c03cdbb3fbf952700ed786e96ccc8b11b861c6df3f8b90fea25a087c,

Merkle Root:

7b4553598407eaf25616da7dbb51a7e45f432b88366615174bc2ee6824991ba3

Transactions: Perso 6 → Perso 4: 0.2 BTC; Perso 2 → Perso 5: 0.5 BTC

Nonce: 40

Hash: 001679ccace0ae9df2a5636fc80d3ba7af94fd58e6e3183ff4f88c80aaaaeed5d

L'erreur se trouve au niveau du Bloc #3. En effet, le hash précédent n'est pas le bon !

2. La règle de la chaîne la plus longue est utilisée

V. Halving et Récompenses de Blocs

Éléments de cours

1. Tous les 210 000 blocs, la récompense est réduite de 50%.
2. Initialement, 50 BTC. Actuellement, 3.125 BTC. Prochain Halving prévu en avril 2028.
3. Le Halving réduit peu à peu la création de nouveaux BTC.
4. Compensation par rapport à la baisse des récompenses.

Questions Pratiques

1. Récompense de 50 BTC: $50 \times 210\,000 = 10\,500\,000$ BTC

2. Récompense de 25 BTC: $25 \times 210\,000 = 5\,250\,000$ BTC;
 $10\,500\,000 + 5\,250\,000 = 15\,750\,000$ BTC
3. Récompense au 4e Halving: 3.125 BTC. Cela représente 6.25% de la récompense initiale.
4. Le dernier Halving date du 20 avril 2024. Depuis, environ 43 000 blocs sont passés.

VI. Principales Attaques

Éléments de cours

1.
 - a. **Attaque 51%**: Scénario dans lequel un acteur malveillant contrôle plus de la moitié de la puissance de calcul du réseau.
 - b. **Double dépense**: Un utilisateur tente de dépenser les mêmes fonds plus d'une fois
 - c. **Sybil Attack**: Attaque par la création de fausses identités pour manipuler le réseau.
 - d. **Transaction malleability (avant SegWit)**: Avant SegWit, l'identifiant des transactions était modifiable ce qui pouvait entraîner des problèmes et des attaques.
2. Les nœuds sont répartis géographiquement, rendant l'acquisition de 51% de la puissance de calcul plus difficile. Cette répartition empêche les attaques ciblées.
3. Un mineur "malhonnête" peut acquérir la majorité du hashrate, effectuer des transactions frauduleuses ou encore faire de la double dépense.

Questions Pratiques

1. BTG en 2018. Symptômes: double dépense et chaîne alternative. Conséquences: perte de fonds, perte de confiance, suspension des transactions
2. XVG en 2018 et ETC en 2019

VII. Exploration d'un Bloc en Ligne

Éléments de cours

1. La hauteur du bloc est son placement dans la chaîne (ex: bloc 10). Le hash est un identifiant unique généré à partir des éléments du bloc et qui dépend donc du contenu, contrairement à la hauteur du bloc.
2. La taille brute était la mesure de la quantité totale de données dans un bloc avant SegWit et le poids (WU) est la nouvelle mesure plus optimisée permettant un poids plus élevé.
3. Il peut y avoir des légers retards sur les systèmes des mineurs et le protocole tolère une légère flexibilité.

Questions Pratiques

1. Bloc #700000
 - a. **Hash:**000000000000000000000000590fc0f3eba193a278534220b2b37e9849e1a770ca959
 - b. **Hauteur:** 700000
 - c. **Timestamp:** 11 sept.2021, 06:14:32
 - d. **Nombre de transactions:** 1 276 ; Taille brute: 1 276 422 ; Weight: 3 998 094 WU
2. Transaction 7
 - a. **TXID:** 7 ; **Inputs:** 1; **Outputs:** 10; **Frais:** 0.00060000 BTC (\$57.72)
 - b. ?

VIII. Frais de Transaction et Mempool

Éléments de cours

1. Frais de transaction en sat = taille en vbytes × taux de frais en sat/vbyte
2. Les mineurs choisissent les transactions à inclure dans le prochain bloc depuis le mempool

3. Le Replace-By-Fee (RBF) est une fonctionnalité du protocole Bitcoin qui permet à un utilisateur de remplacer une transaction non confirmée par une autre transaction avec des frais plus élevés afin d'augmenter les chances que la transaction soit incluse rapidement dans un bloc.
4. La taille, des scripts complexes, le fait d'utiliser SegWit plutôt que Legacy sont des facteurs pouvant influencer la confirmation d'une transaction

Questions Pratiques

1.

id	taille (vbytes)	fee rate (sat/vbyte)	Frais de transaction (sat)	Ordre
TxA	300	10	3000	2
TxB	200	15	3000	1
TxC	100	5	500	3
2. Transaction:
2996591ec2bd26d5d33b5c2d3c480ddb45c97e39cb85273b95234a491c889edb
Taille: 222 bytes
Frais transaction: 5640 sat
Fee rate = Frais transaction / Taille = 25.405 sat/bytes
3. Grâce au Replace-By-Fee, on peut soumettre une nouvelle transaction avec des frais plus élevés.

IX. Recherche Complète d'un Bloc

Éléments de cours

1. Pour décrire un bloc, on observe sa hauteur, sa version, sa difficulté, son merkle root, le nombre de transactions, l'horodatage...
2. Grâce à l'adresse de sortie où il reçoit la récompense ou encore des tags comme un nom de pool de minage.
3. La version peut changer si le protocole et les règles changent ou évoluent.

Questions Pratiques

1. BLOC #500000
 - a. **Minage:** BTC.com
 - b. **Version:** 0x20000000
 - c. **Bits:** 402 691 653 / **Difficulté:** 1 873 105 475 221,61
 - d. **Nombre de transactions:** 2 701
2. BLOC #700000
 - a. Il y a moins de transactions dans le bloc 700000 (1276)
 - b. Le bloc 500000 a été miné le plus rapidement
 - c. Moins de transactions, plus de temps de minage ?

X. Sécurité et Décentralisation (Nœuds)

Éléments de cours

1. Un nœud complet contient une copie complète de la blockchain et valide les transactions indépendamment des autres nœuds.
2. Un nœud SVP télécharge une partie de la blockchain: les en-têtes des blocs, les merkle proofs et les transactions spécifiques à vérifier.
3. La multiplication et la dispersion géographique des nœuds rendent le réseau résistant aux attaques et à la censure, sécurisé, stable, libre et accessible à tous.
4. Les mineurs proposent et minent les blocs, les nœuds complets décident de les valider ou non.

Questions Pratiques

1. Bitnodes.io
 - a. 21 827 nœuds sont enregistrés actuellement.
 - b. La majorité n'est pas renseignée. Sinon, le pays où il y a le plus de nœuds sont les Etats-Unis. Il y en a aussi beaucoup en Europe.
2. Non, je ne serais pas prête à héberger un full node. Malgré les avantages tels que la sécurité, la confidentialité et l'anonymat, héberger un full node nécessite beaucoup de ressources matérielles et consomme énormément de bande passante.
3. Via la preuve de travail (proof of work). Il fait confiance à la chaîne la plus longue et la plus dure à falsifier. Il peut vérifier une transaction et vérifier son merkle root.

XI. Hard Fork vs Soft Fork

Éléments de cours

1. Un hard fork est une mise à jour du protocole qui n'est pas compatible avec les anciennes versions, contrairement au soft fork qui est compatible.
2. **Exemple de hard fork:** ETC ; **Exemple de soft fork:** SegWit
3. Il y a une séparation en deux chaînes distinctes, suivant des règles différentes.
4. Séparation du réseau, création d'une nouvelle blockchain et potentiellement une nouvelle crypto-monnaie

Questions Pratiques

1. Bitcoin Cash a été créé le 1er août 2017. Les arguments des partisans de cette scission concernent une augmentation de la taille des blocs proposée par ce hard fork. Le but était de réduire les frais et de rendre les transactions plus rapides.
2. SegWit a été activé le 24 août 2017 (Bloc #481 824). Les données de signature sont maintenant séparées des données principales des transactions et il y a un nouveau format d'adresse.
3. Un soft fork mal conçu pourrait involontairement devenir un hard fork et créer des divergences dans le réseau.

XII. SegWit et la Notion de "Weight"

Éléments de cours

1. SegWit est une mise à jour du protocole Bitcoin (soft fork). Avant SegWit, un attaquant pouvait modifier une transaction avant sa confirmation en changeant la signature.
2. Le poids est calculé par $\text{Weight} = (\text{Taille du bloc sans témoin} \times 3) + (\text{Taille totale du bloc})$. La limite d'un bloc était initialement de 1 MB soit 1 000 000 d'octets et la nouvelle limite est de 4 000 000 WU ce qui s'apparenterait à 4 MB.
3. Avec SegWit, les signatures sont séparées des données principales du bloc et ne comptent plus de la même manière dans la taille du bloc.

Questions Pratiques

1. Bloc #521096
 - a. **Taille:** 1 058 308 octets ; **Poids:** 3 993 103 WU. On est presque à la limite de poids mais loin d'une limite de 4 MB.
 - b. Transaction de hash
e714a8eda9c38df2196d96bff628e699fc7918d4e1b0e50f336c7e8a2f9ba04b.
Par rapport à une transaction legacy, une nouvelle section "Témoin" (Witness) apparaît.
 - c. La signature est stockée dans un groupe à part, et n'influence donc plus autant la transaction
 - d. Complication inutile, préférence pour Bitcoin Cash...

XIII. Clés Privées, Clés Publiques, Adresses et WIF

Éléments de cours

1. ?
2. L'adresse Bitcoin est un identifiant dérivé de la clé publique qui permet de recevoir des paiements. Les différents formats sont Legacy (P2PKH), P2SH et Bech32
3. Le format est plus court et lisible, a une bonne compatibilité et une flexibilité.
4. Perte des fonds ou vol des fonds

Questions Pratiques

1. **Clé privée:**
1E99423A4EDF94A41D2FDE4FBC7E4A9DAA39CCAF48FA660B47CAED42E34A4B5A
 - a. **WIF:** L1aW4aubDFB7yfras2S1mMEG9JwRQDVr3X9y5SoF7bG2x6Vr3muE
 - b. **Adresse P2SH:** 3J98t1WpEZ73CNmQviecrnyiWrnqRhWNLy

XIV. Introduction au Lightning Network

Éléments de cours

1. La lenteur et les frais élevés
2. On ouvre le canal, on effectue autant de transactions hors chaîne que l'on veut, on ferme le canal en publiant la dernière transaction sur la blockchain.
3. Pour réduire la congestion
4. Le réseau forme des chemins interconnectés de canaux de paiement entre plusieurs personnes.

Questions Pratiques

1. Les paiements peuvent être routés à travers plusieurs nœuds intermédiaires, chacun ayant un canal avec les nœuds voisins. Le réseau calcule automatiquement un chemin optimisé pour que le paiement atteigne sa destination, en utilisant les canaux ouverts avec d'autres utilisateurs.
2. Pour des transactions rapides et de petite taille.

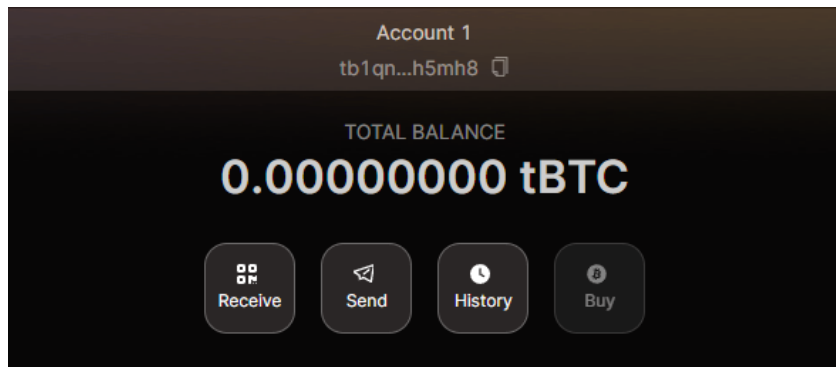
XV. Installation d'un Wallet et Transactions sur le Testnet

Éléments de cours

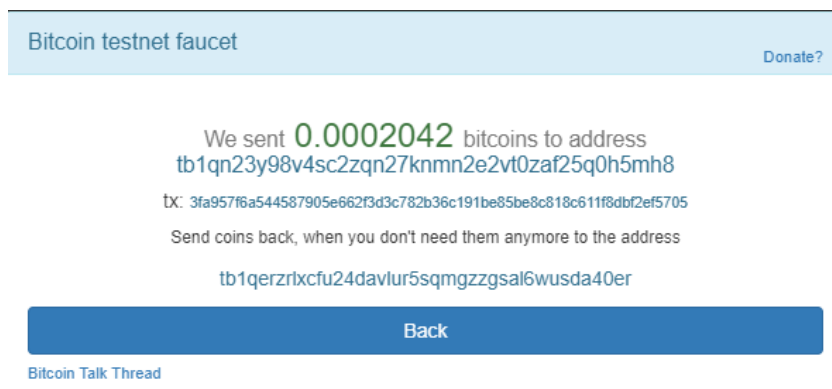
1. Le mainnet est le réseau principal, la version officielle et active de la blockchain tandis que le testnet est le réseau dédié aux tests.
2. On peut utiliser le Testnet pour apprendre à utiliser la blockchain sans risque financier par exemple.

Questions Pratiques

1.

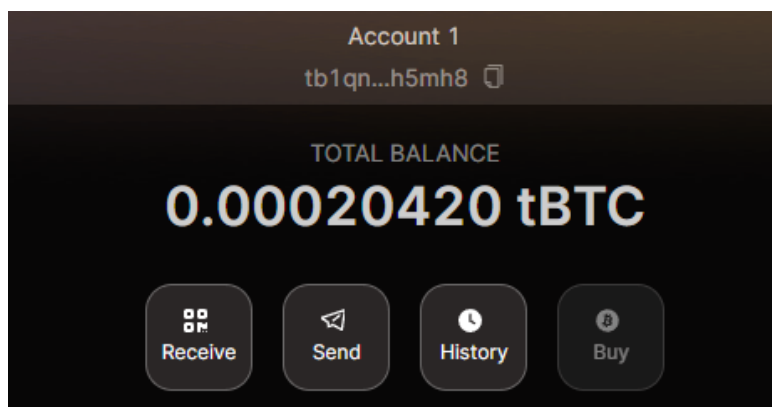


2.

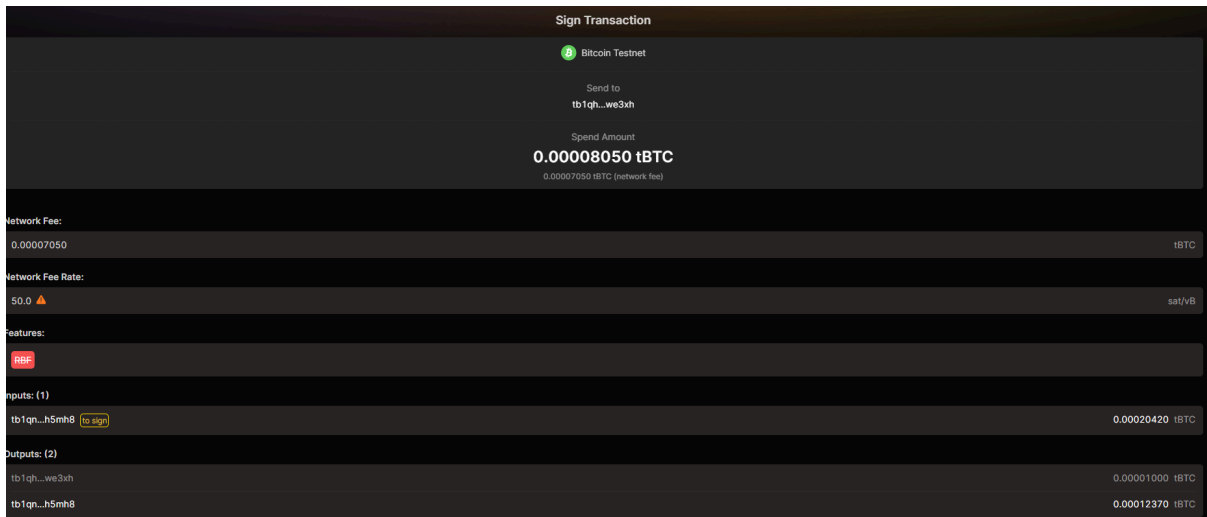


3.

a.



b. c.



Sign Transaction

Bitcoin Testnet

Send to
tb1qh...we3kh

Spend Amount
0.00008050 tBTC
0.00007050 tBTC (network fee)

Network Fee:
0.00007050 tBTC

Network Fee Rate:
50.0 sat/vB

Features:
RBF

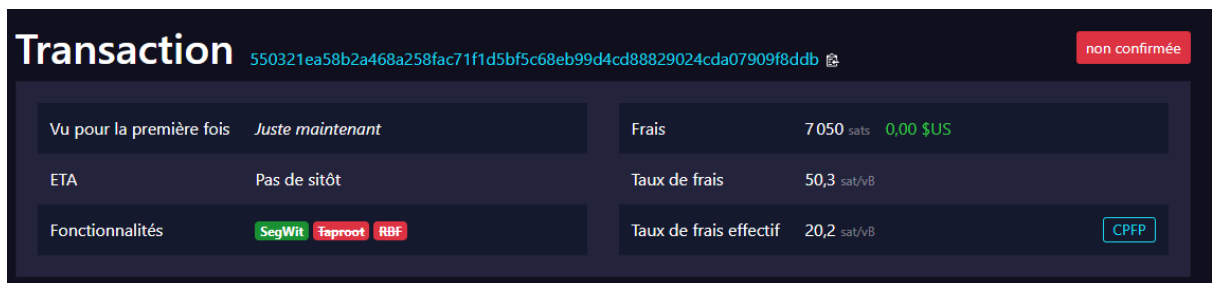
Inputs: (1)
tb1qn...h5mh8 (to sign) 0.00020420 tBTC

Outputs: (2)
tb1qh...we3kh 0.00001000 tBTC
tb1qn...h5mh8 0.00012370 tBTC

d.

550321ea58b2a468a258fac71f1d5bf5c68eb99d4cd88829024cda07909f8ddb

4.



Transaction 550321ea58b2a468a258fac71f1d5bf5c68eb99d4cd88829024cda07909f8ddb non confirmée

Vu pour la première fois	Juste maintenant	Frais	7 050 sats 0,00 \$US
ETA	Pas de sitôt	Taux de frais	50,3 sat/vB
Fonctionnalités	SegWit Taproot RBF	Taux de frais effectif	20,2 sat/vB CPFP

5. Questions/Réflexions

- Avantages: accessibilité, pas de grosse installation // Inconvénients: moins sécurisé, moins bon stockage à long terme
- J'ai pu comprendre comment fonctionnait un wallet, de sa création jusqu'à son utilisation et comprendre comment étaient effectuées les transactions.