Saito Consensus

Fixing the Market Failures in Bitcoin

David Lancashire | Richard Parris

24 DÉCEMBRE 2020 | V. 4.0.1
TRADUCTION FRANÇAISE | V. 1.2

Saito

saito.io

Résumé.

SAITO résout les problèmes d'action collective qui empêchent l'extensibilité (scalability) dans les blockchains de type preuve de travail (PoW – Proof of Work) et preuve d'enjeu (PoS – Proof of Stake) en couplant un registre circulaire à un mécanisme de consensus qui encourage la collecte et le partage des frais de transaction. Le réseau résultant, ne paie pas seulement le minage et la mise en garantie (staking), mais pour toutes les activités qui contribuent à la valeur économique du réseau. Dans ce processus, SAITO élimine complètement l'attaque des 51 %, entre autres...

Saito

Saito paie les nœuds d'infrastructure orientés vers l'utilisateur à partir d'un mécanisme de consensus ouvert. Comme cette approche évolue naturellement (extensibilité), elle incite le traitement de grandes quantités de données et peut être utilisée pour créer des versions décentralisées/ouvertes de nombreux services à fort débits de données, tels que les échanges de données sans possibilité de diffusion, les applications d'authentification et de monétisation, les registres de clés distribués qui sont à l'abri des attaques MITM, les canaux de paiement, et bien plus encore.

En termes économiques, Saito peut être assimilé comme étant une solution pour inciter le marché libre à fournir un bien public. Sa conception corrige les problèmes d'action collective inhérents aux mécanismes PoW et PoS, de sorte que les participants à la recherche de profits sont en concurrence pour apporter de l'argent dans le réseau, ce qui permet une extensibilité au point que, le matériel de réseau sous-jacent plutôt que les contraintes économiques, imposent des limites à la croissance de la chaîne de blocs (blockchain). Nous pensons actuellement que la limite pratique pour une blockchain Saito est de l'ordre de 100 TB de données par jour, et les progrès en matière de capacité de routage nous pousseront au niveau du pétaoctet dans une décennie environ.

La section suivante décrit brièvement les problèmes économiques qui doivent être résolus afin de construire une blockchain extensible (scalable). Les sections suivantes décrivent la façon dont Saito résout ces problèmes et détaillent l'implémentation de ces méthodes.

100TB / Jour

⌄

XPB / Jour

Saito

Le problème.

Le problème d'extensibilité d'une blockchain ne se situe pas au niveau de la technologie du réseau : au moment où nous écrivons ces lignes, les centres de données du monde entier mettent en œuvre des commutateurs de réseau de 400 Gbps tandis que les connexions de 100 Gbps deviennent la norme, même dans les installations mutualisées de niveau inférieur. Si nous avions les ressources pour payer l'équipement nécessaire il n'y a rien qui nous empêche techniquement de construire une blockchain qui est aussi décentralisée et ouverte que l'infrastructure de l'Internet

Ce qui limite la croissance du réseau est le défi de payer pour le réseau. Dans le passé, les non-économistes ont ignoré cette limitation, en prétendant que tant que quelqu'un gagnait de l'argent avec le réseau, il paierait tous les coûts nécessaires pour le soutenir. Mais ce n'est pas vrai, car les réseaux PoW et PoS sont affligés par deux défaillances du marché : une tragédie des biens communs qui engendre la saturation et l'effondrement éventuel de la blockchain, et un problème de resquillage (free-riding) qui conduit à une sous-provision des infrastructures réseau pour l'utilisateur ainsi que d'une surproduction d'activités payantes comme le minage et la mise en garantie (mise en jeu / *staking*). Aucun de ces deux problèmes n'est paralysant à petite échelle, mais ils deviennent incapacitants à mesure que la bande passante et le stockage coûtent de l'argent.

Existe-t-il des alternatives ? Face à la nécessité de payer pour une infrastructure réseau non compensée, les développeurs reposent le problème sur le marché. Comme les économistes le savent depuis les années 1960, demander au secteur privé de financer une infrastructure non compensée nécessite une restriction quelque part dans le modèle économique. Les entreprises qui paient pour la collecte des frais doivent nécessairement fermer l'accès aux droits qu'elles reçoivent. Le contrôle (centralisation / privatisation) du flux de fonds dans la blockchain affaibli alors l'ouverture (décentralisation / source libre) de la couche de consensus.

La seule solution viable est d'éliminer ces défaillances du marché au niveau des mesures incitatives. Avant que la solution ne puisse être comprise, elle doit être vue clairement. Les principaux problèmes sont les suivants.

Le problème de la tragédie des biens communs (Tragedy of the commons) née de l'existence du registre permanent, qui encourage les nœuds à accepter d'être payés aujourd'hui pour un travail qui peut être transféré à d'autres le lendemain. Cet incitatif conduit à des chaînes de blocs saturées et, plus subtilement, à une mauvaise tarification des transactions, car les utilisateurs peuvent payer des frais qui ne reflètent pas le coût réel de leur transaction pour l'ensemble du réseau. Le fait qu'il s'agisse d'un problème fondamental est évident du fait que la solution de Satoshi est de "ne pas s'en soucier", une approche qui n'est plus viable dans les réseaux qui fonctionnent à une grande échelle au niveau économique.

L'élimination du problème de la tragédie des biens communs exige que tous les nœuds qui ajoutent des transactions à la blockchain supportent le coût du traitement de ces transactions aussi longtemps qu'elles restent sur la blockchain. En pratique, cela nécessite un mécanisme de marché qui détermine avec précision le prix du stockage des données sur la chaîne. Il faut également élimer la dérive de la blockchain ou de différer la perception des frais afin que les paiements soient effectués au fur et à mesure que le nœud continue d'effectuer le travail nécessaire pour le paiement.

Notre solution à ce problème est décrite dans la section 2.

Le problème.

Le problème du resquillage (free-riding) est plus insidieux. Il apparaît dans les blockchains où les paiements sont effectués pour un type de travail spécifique (comme le minage ou la mise en garantie) au détriment de d'autres activités nécessaires. Ce décalage incite les participants à maximiser leurs dépenses en travail rémunéré et à minimiser leurs dépenses sur toute autre activité. Dans l'espace blockchain, cela se traduit par des mineurs et des *stakers* profitant de ceux qui font le travail non rémunéré de la collecte des frais, de développer des applications ou de soutenir autrement le réseau.

Le problème s'aggrave lorsque le réseau s'étend, et les pressions de l'extensibilité rendent le piège inéluctable : tout mineur de bitcoin qui dépense un pourcentage de ses revenus plus petit que ses collègues altruistes pour le hachage, perdra des parts de marché jusqu'à ce qu'il capitule aussi. En économie, la solution typique au problème du resquilleur (free-rider) est d'éliminer la propriété de non-exclusion associée à tout bien ou service : restreindre les avantages à ceux qui paient les coûts de prestation. Dans le monde de la blockchain c'est impossible à entreprendre, sans détruire l'ouverture (décentralisation) du réseau. Les développeurs abordent souvent le problème en ajoutant une couche intermédiaire de protection, par exemple en enveloppant les consensus de paiements dans des cercles de vote fermés qui sont bien sûr eux-mêmes sensibles à ces attaques.

Sans une solution à ce problème, notre choix est entre un réseau qui ne peut pas évoluer parce qu'il ne peut pas payer l'activité du réseau, ou un réseau qui change d'échelle mais perd son ouverture, la confiance des utilisateurs et son autosuffisance économique qui rendent l'invention de la blockchain utile. Aucune des deux approches n'est utile pour construire une blockchain véritablement ouverte à grande échelle. La solution théorique au problème du resquilleur nécessite d'éliminer la possibilité de celui-ci : fixer la structure sous-jacente des incitatifs, de sorte que les participants soient payés pour fournir ce dont le réseau a réellement besoin. Parce que la blockchain nécessite un coût quantifiable de l'attaque, ceci nécessite l'élimination du minage et la mise en garantie, et de passer à une forme différente de travail qui mesure et paie les nœuds proportionnellement à la valeur qu'ils fournissent au réseau plutôt qu'à la quantité de hachage ou de *staking* qu'ils effectuent.

Cela nécessite que nous trouvions une nouvelle façon de mesurer la valeur et ensuite de payer les nœuds proportionnellement à la quantité de valeur qu'ils apportent. Pour y parvenir il faut dériver notre mesure du travail à partir des frais de transaction que les utilisateurs paient. Le travail de l'acheminement des frais de transaction dans le réseau est le travail que notre réseau doit encourager. Les nœuds honnêtes peuvent être incités à faire ce travail par une part des frais collectés. La difficulté se déplace pour s'assurer que ce mécanisme préserve les propriétés de coût d'attaque, de telle sorte que les attaquants ne puissent pas dépenser leur propre argent pour attaquer le réseau, et le récolter à nouveau dans une boucle perpétuelle.

Le mécanisme de sécurité décrit dans la section 3, expose une méthode technique permettant d'atteindre cet objectif.

Résoudre la TRAGÉDIE DES BIENS COMMUNS.

Saito résout le problème des blockchains en permettant aux nœuds du réseau de supprimer les blocs les plus anciens dans le registre à des intervalles prévisibles (*epoch* / période).

La durée des *epoch* est spécifiée dans le code du consensus. Un cas extrême - une blockchain conçue pour un trafic de niveau mondial pour des applications d'échange distribués de clés, peut avoir une *epoch* aussi courte que 24 heures.

Saito précise qu'une fois qu'un bloc tombe en dehors de l'*epoch* actuelle, ses sorties non dépensées sont supprimées (UTXO – en angais) et ne sont plus dépensables. Mais tout UTXO de cet ensemble qui contient suffisamment de jetons pour payer des frais de rediffusion doit être réinclus dans le bloc suivant. Les producteurs de blocs font cela en créant une "rediffusion automatique des transactions" (Automatic Transaction Rebroadcasting / ATR) qui comprennent les données de la transaction originale, mais qui ont un nouvel UTXO. Après deux *epoch*, les producteurs de blocs peuvent supprimer toutes les données du bloc, bien que le hachage de 32 octets de l'en-tête peut être conservé pour prouver la connexion avec le bloc de original (Genesis block).

Le mécanisme ATR résout complètement le problème de la tragédie des biens communs, rendant impossible pour une blockchain de devenir si grosse qu'elle s'effondre. La clé est de s'assurer que la "redevance de rediffusion" payée par les transactions ATR soit un multiple positif de la moyenne des frais payés par les nouvelles transactions au cours l'*epoch* précédente. Au fur et à mesure que la blockchain se développe et qu'il y a moins d'espace disponible pour les nouvelles transactions, la concurrence sur le marché pousse les frais payés par les nouvelles transactions à la hausse. Cela pousse à la hausse les frais payés par les transactions plus anciennes et augmente la quantité de données élaguées par la blockchain. Le marché atteint l'équilibre où les anciennes données sont retirées de la chaîne au même rythme que les nouvelles données sont ajoutées. La découverte par le marché du coût réel du traitement de la blockchain est un effet secondaire de cette structure incitative, qui fonctionne en éliminant l'incitatif des producteurs de blocs qui vont rajouter des données non rentables à la chaîne. Ce mécanisme permet d'éviter les problèmes liés au fait que les développeurs codent des variables économiques et donc empêche les formes subtiles de resquillage (free-riding) que l'on trouve dans d'autres chaînes (suppression de données sur la chaîne, refus de stocker ou de partager des blocs historiques) où l'élagage est effectué pour économiser de l'argent. Toutes ces formes de tricherie disparaissent car les nœuds qui ne stockent pas la totalité de la blockchain sont incapables de produire de nouveaux blocs, car ils ne savent pas quels paiements doivent être rediffusés.

Cela évite le problème de notre blockchain grandissant trop rapidement pour que les nœuds du réseau puissent la stocker, et garantit que l'espace sur la blockchain peut être tarifé avec précision, même si les temps de stockage approchent l'infini, fixant ainsi la tragédie des biens communs qui ne permet pas de rapporter de l'argent aux nœuds du réseau pair-à-pair qui paient pour toutes les activités diverses permettant au réseau de fonctionner. **Pour résoudre ce problème, nous avons besoin d'un nouveau mécanisme de consensus.**

Saito

Éliminer le RESQUILLAGE.

Dans Saito, tout nœud peut créer un bloc à tout moment à condition qu'il ait suffisamment de travail de routage dans son *pool* de mémoire. La quantité de travail de routage nécessaire à la production d'un bloc dépend de la la rapidité avec laquelle un bloc suit son prédécesseur : les règles du consensus augmentent la valeur immédiatement après la découverte d'un bloc et la diminue progressivement jusqu'à ce qu'elle atteigne zéro. Puisque les producteurs de blocs émettent des blocs dès qu'ils deviennent profitables, le rythme de production des blocs est déterminé par la quantité globale de travail de routage générée par le réseau.

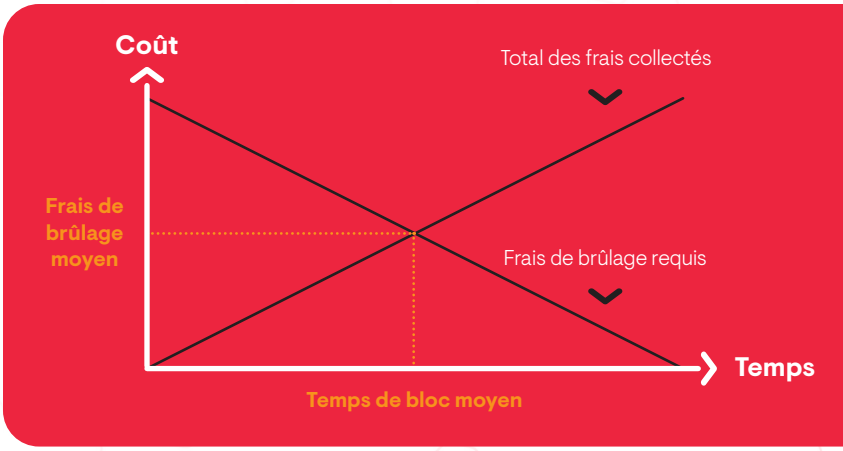


Schéma 1 ▸ Frais de brûlage de jeton

Saito tire son travail de routage des frais de transactions intégrés dans chacune de celle-ci. L'utilisation de cette mesure du travail pour produire des blocs, rend l'attaque du réseau onéreux, puisque la réorganisation coûtent de l'argent. On peut voir sur le schéma 2 qu'il est impossible aux attaquants de produire des blocs à un rythme plus rapide que la chaîne principale, sauf s'ils ont accès à un plus grand nombre de frais de transaction.

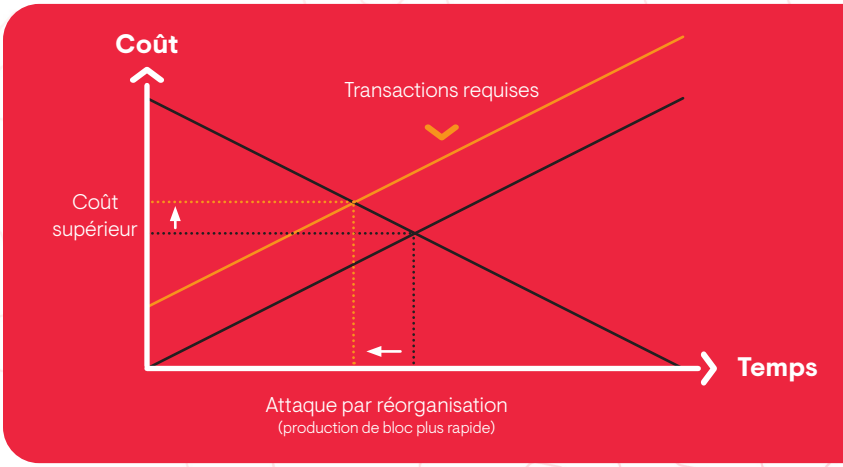


Schéma 2 ▸ Coût des frais de brûlage

Saito

Éliminer le RESQUILLAGE.

Pour sécuriser ce mécanisme, Saito fait en sorte que les nœuds de routage signent cryptographiquement les transactions à mesure qu'elles se propagent dans le réseau. Les règles du consensus spécifient que la quantité de travail de routage qu'une transaction fournit à un nœud diminue avec le nombre de sauts dans son chemin de routage, et que les transactions ne fournissent aucun travail de routage utilisable aux nœuds qui ne sont pas dans leur chemin de routage. Le travail utilisé pour produire des blocs est la collecte et le partage efficaces des frais de réseau entrants.

Tant qu'il n'y a pas de paiement pour la production de blocs, ce système offre une sécurité comparable à celle de Bitcoin : le coût d'une attaque est toujours quantifié et les attaquants doivent dépenser leur propre argent pour attaquer la chaîne. Cela permet aux utilisateurs d'atteindre le nombre de confirmations nécessaires pour répondre à leurs exigences de sécurité. En prime, le réseau peut augmenter la quantité de travail de routage nécessaire à la production de blocs afin de maintenir un temps de bloc constant quand le volume de transactions augmente, de sorte que la sécurité s'adapte avec le volume des transactions.

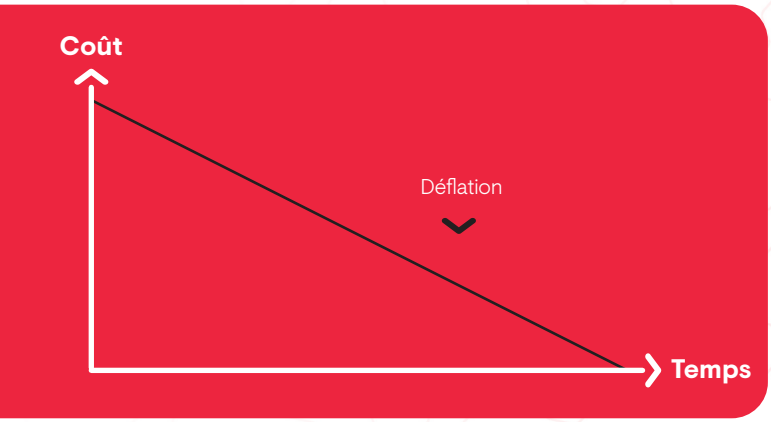


Schéma 3 ▸ Déflation des frais de brûlage dans le temps

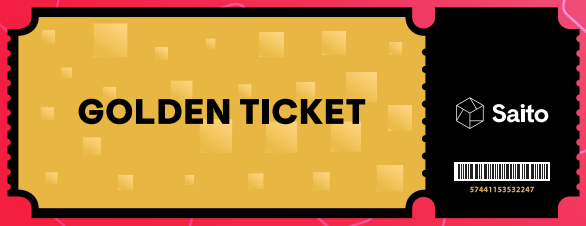
Le problème majeur de cette approche réside dans les conséquences de l'obligation pour le réseau de brûler du capital pour produire des blocs.

Pour éviter un crash déflationniste, il faut réinjecter des jetons dans notre réseau. Mais Saito ne peut pas simplement donner les frais directement aux producteurs de blocs : cela permettrait aux attaquants d'utiliser les revenus d'un bloc pour générer le travail de routage nécessaire pour produire le suivant. Diviser le paiement entre les différents nœuds est préférable, mais tant que les producteurs de blocs exercent une influence sur qui doit être payé, un attaquant astucieux peut sybiliser le réseau ou mener des opérations de broyage qui ciblent le mécanisme d'émission de jetons.

Pour résoudre ce problème, il faut inverser la solution classique de preuve de travail. Dans Bitcoin, les règles de consensus rendent la production de blocs coûteuse et les frais sont alors remis au producteur du bloc. Ceci est sensé garantir que la production de blocs est coûteuse mais en réalité, cela garantit qu'il y a toujours des conditions dans lesquelles les attaques sont profitables.

Dans le cas de Saito, la solution est inverse. Le problème de premier ordre se transforme en sécurisation du mécanisme de paiement : garantir que les paiements sont proportionnels au travail, indépendamment de qui produit le bloc. Le coût de l'attaque que crée un mécanisme avec cette propriété est alors transformée en étant le coût de production des blocs. Nous appelons le mécanisme qui atteint cet objectif "la solution du ticket d'or". Ce mécanisme paie les nœuds honnêtes pour la collecte des frais indépendamment de la source de production des blocs. L'astuce, est de tirer cela d'une manière qui garantit qu'il y a toujours un coût quantifiable pour attaquer le système. La solution pratique est de retourner les frais de transaction au réseau à travers un processus qui ne peut être manipulé par aucun des acteurs du réseau sans dépenser beaucoup plus d'argent pour l'attaque que ce qu'ils n'ont à gagner en collectant les paiements. Les détails de l'implémentation sont décrits dans la section suivante.

Le ticket d'or.



Chaque fois qu'un nœud produit un bloc, il peut collecter la différence entre la quantité de travail de routage inclus dans son bloc et celle requise pour la production du bloc. Aucun autre paiement n'est effectué.

Pour débloquer ces paiements, le réseau doit se résoudre à un puzzle informatique que nous appelons le "ticket d'or". Ce puzzle nécessite la connaissance du bloc de hachage pour le résoudre et ne peut pas être calculée à l'avance. Les mineurs du réseau surveillent les blocs lorsqu'ils sont produits et commencent le hachage à la recherche d'une solution. S'ils trouvent une transaction, ils la propagent à nouveau dans le réseau comme une transaction payante normale.

Une seule solution peut être incluse dans un bloc, et cette solution doit être incluse dans le bloc suivant pour être considérée comme valide. Si ces conditions sont violées ou si un ticket d'or n'est pas résolu, les fonds qui n'ont pas été payés dans le bloc précédent ne sont tout simplement pas alloués. Ils reviennent en arrière dans la chaîne et finissent par se retrouver en dehors de celle ci, et à ce moment-là les fonds perdus sont récupérés par la couche de consensus et finalement redistribués dans le cadre d'une future récompense de bloc.

Si une solution est trouvée à temps, les frais non alloués sont libérés dans le réseau ; répartis entre le mineur qui a trouvé la solution et un nœud aléatoire dans le réseau de routage pair-à-pair. Le nœud de routage chanceux est sélectionné en utilisant une variable aléatoire provenant de la solution du mineur, où la chance de gagner pour chaque nœud de routage étant normalisée pour être proportionnelle au travail de routage global qu'il a contribué à la résolution du bloc.

Au fur et à mesure que les transactions sont acheminées dans le réseau, on peut voir que le total des réclamations sur les blocs est plus élevé (somme du travail de routage disponible pour tous les nœuds dans le chemin de routage) augmente tandis que la quantité de travail disponible pour produire un bloc (la quantité de travail de routage disponible pour certains spécifiques) réduit. Les attaquants qui utilisent des transactions honnêtes pour produire des blocs se mettent dans l'obligation de payer la différence.

Nous appelons la répartition du paiement entre les mineurs et les routeurs, le "PaySplit" du réseau. Elle est fixée par défaut à 0,5 (moitié pour les mineurs, moitié pour les routeurs), mais peut être ajustée comme décrit dans la section ci-dessous. Le système du ticket d'or peut être visualisé comme suit :

Schéma 4 ▸ Le système du Ticket d'or

Saito

Le ticket d'or

Ce système présente plusieurs avantages majeurs par rapport aux mécanismes de preuve de travail et de preuve d'enjeu. Le plus important est le fait que Saito distribue explicitement les frais aux nœuds qui fournissent un service aux utilisateurs, collectent les transactions et produisent les blocs, et le fait proportionnellement à la quantité de la valeur que ces acteurs apportent à l'ensemble du réseau. Les nœuds du réseau sont en concurrence pour l'accès au flux lucratifs de transactions entrants, et financeront volontiers toutes les activités de développement nécessaires pour attirer les utilisateurs sur le réseau.

Il convient de noter que les services fournis par les nœuds de passerelle (edge nodes) pour amener à l'utilisation de Saito peuvent inclure l'infrastructure publique requise par d'autres blockchains.

Il s'agit d'un changement fondamental. Là où d'autres blockchains définissent explicitement quelles activités ont de la valeur, Saito laisse les utilisateurs signaler quels services fournissent de la valeur par le coût des frais d'utilisation, tandis que le réseau déduit qui mérite d'être payé. Saito encourage également les participant à fournir un service efficace aux utilisateurs. En payant pour la valeur plutôt que pour un sous-ensemble d'activités du réseau, il fournit le seul moyen de garantir qu'un réseau autosuffisant puisse rester ouvert (décentralisé) et économiquement indépendant à grande échelle.

Le mécanisme de consensus de Saito est également deux fois plus sécurisé que les mécanismes PoW et PoS. Les nœuds honnêtes acheminent les transactions aux producteurs de blocs et gagnent une redevance en échange. Mais les attaquants sont pris dans un cercle vicieux : ils doivent non seulement dépenser le même montant de frais que le réseau honnête, pour produire une chaîne compétitive, mais ils doivent aussi s'aligner sur 100 % de la production minière pour trouver suffisamment de solutions de tickets d'or pour récupérer leurs fonds. Même si les attaquants réussissent à lancer des attaques de récupération de frais, ils doivent toujours dépenser 100 % de leurs revenus pour le hachage.

La version de base du système Saito atteint 100 % de frais de sécurité, éliminant complètement l'attaque des 51%. La section 5 décrit une modification de ce mécanisme qui pousse la sécurité au-delà des 100% et garantit que les attaquants perdront de l'argent en toutes circonstances.

Quelle que soit l'implémentation utilisée, les problèmes économiques créés par les mécanismes qui reposent sur des courbes d'offre externes disparaissent : le minage sert de fonction de coût pur au lieu de fonction de difficulté et la blockchain reste sécurisée même si la courbe d'offre de la puissance du hash devient parfaitement plate.

La version de base du système Saito atteint 100% de frais de sécurité

La sécurité avancée POWSPLIT.

Il est possible d'augmenter les coûts d'attaque au-delà de 100 % des rendements disponibles grâce à un mécanisme de "PowSplit". Notez que dans l'implémentation normale de Saito avec un "PaySplit" fixe de 0,5, le réseau ajuste automatiquement la difficulté d'extraction de sorte qu'en moyenne, une solution soit trouvée par bloc. Comme les mineurs ne peuvent pas contrôler la variance à laquelle les solutions sont trouvées, la difficulté du réseau peut finir par être, en moyenne, plus faible que nécessaire pour une sécurité optimale.

Une approche "PowSplit" élimine ce problème en ajustant la difficulté d'extraction de manière qu'une solution soit en moyenne trouvée tous les N blocs. Lorsqu'une telle solution est incluse dans la blockchain, si le bloc précédent ne contenait pas de ticket d'or, la variable aléatoire utilisée pour choisir le nœud de routage gagnant est hachée à nouveau pour sélectionner un gagnant à partir du bloc non résolu qui l'a précédé ou à partir d'une table de *stakers* comme décrit ci-dessous. Une limite supérieure à la récursion inverse peut être appliquée à des fins pratiques, car la boucle de la blockchain récupérera tous les fonds qui ne sont pas payés.

Pour devenir un *staker* dans le réseau, les utilisateurs diffusent une transaction contenant un UTXO spécialement formaté. Ces UTXO sont ajoutés à une liste de *stakers* en attente lors de leur inclusion dans un bloc. Une fois que la table de mise en garantie courante a été entièrement payée, tous les UTXO en attente sont déplacés dans la table courante. Pour éviter les attaques par "étranglement" (saturation) sur le mécanisme de mise en garantie (staking), il est sage de ne pas permettre aux *stakers* de retirer ou de dépenser les UTXO engagés avant d'avoir reçu le paiement.

Le pourcentage du revenu du réseau alloué aux nœuds de *staking* doit être proportionnel à leur part du montant des frais payés dans le réseau par le mécanisme de mise en garantie au cours du tour précédent. Des limites peuvent être imposées à la taille du *pool* de mise en jeu afin de créer une concurrence entre les *stakers* si cela est souhaitable ou d'empêcher les utilisateurs de spammer le mécanisme de mise en jeu dans l'espoir de dissuader les *stakers* honnêtes de participer. Dans des situations normales, la boucle de la blockchain et le mécanisme ATR empêcheront les *stakers* de lancer des attaques de spam, car les multiples UTXO paieront tous des frais de rediffusion.

Pour garantir le fonctionnement du système, les producteurs de blocs qui rediffusent des UTXO doivent désormais indiquer dans leurs transactions ATR si les sorties spécifiques sont actives dans le *pool* de mise en jeu actuel ou en attente. Une représentation hachée de l'état de la table de mise en jeu peut être incluse dans chaque bloc sous forme d'engagement pour permettre aux nœuds de vérifier l'exactitude de la table, mais le mécanisme de rediffusion ATR permettra théoriquement à tous les nœuds de reconstruire l'état du *pool* en une *epoch* (période) au maximum.

La difficulté d'extraction peut être ajustée à la hausse si deux blocs contenant des tickets d'or sont trouvés à la suite et à la baisse si deux blocs sans tickets d'or sont trouvés consécutivement. Un coût punitif similaire peut étrangler le paiement de la mise en jeu si deux blocs sans ticket d'or sont trouvés consécutivement (un montant toujours plus important du revenu de la mise est retenu). Nous encourageons les personnes intéressées par les problèmes mathématiques qui en découlent à consulter nos articles sur le sujet. Le coût d'attaque d'un réseau basé sur le mécanisme de Saito dépasse largement les 100 %.

Saito

La sécurité avancée PAYSPLIT.

Il y a plusieurs modifications au mécanisme du PaySplit qui peuvent être utilisées pour augmenter les coûts de l'attaque. Cette section décrit une amélioration théorique qui permet de faire "flotter" le PaySplit et qui fonctionnera sous certaines hypothèses concernant la rationalité du réseau.

Une implémentation de ce système modifie les blocs de façon à ce qu'il incluent un vote pour augmenter, diminuer ou maintenir constamment le PaySplit du réseau.

Les solutions des tickets d'or peuvent alors être modifiées afin pour qu'elles contiennent un vote similaire sur la difficulté de la fonction de production de ce dernier.

Ajuster le PaySplit de cette manière peut changer la distribution des frais de transaction entre les nœuds de routage et les mineurs en temps réel. Cela permet au réseau d'atteindre un équilibre optimal plutôt qu'un équilibre arbitraire. Pour éviter que cet équilibre ne reflète uniquement les préférences des nœuds de routage et des nœuds d'extraction, nous recommandons de laisser aux utilisateurs du réseau d'annoter leurs transactions avec un vote optimal pour le PaySplit : si une transaction provenant d'un utilisateur contient un tel vote, elle ne peut qu'être incluse dans un bloc qui vote dans la même direction. Les utilisateurs qui prennent parti dans la lutte entre routeurs et mineurs sacrifient ainsi la fiabilité et la rapidité de la confirmation des transactions, mais gagnent une influence marginale sur la façon dont le réseau distribue les frais. Les utilisateurs qui votent, retiennent également leurs frais pour les nœuds votant différemment d'eux.

Dans des conditions où les participants au réseau font preuve d'une rationalité limitée, ce mécanisme pousse le PaySplit jusqu'au point où la sécurité fournie est optimale pour tous les participants sachant le coût de la collecte des frais supplémentaires. Le *Contrat social* d'Alexis De Tocqueville confirme cet équilibre : deux participants quelconques dans la structure de réseau tripartite (routeurs, mineurs, utilisateurs) peuvent faire équipe pour ramener le PaySplit vers l'idéal souhaité. Nous laissons la recherche sur ce mécanisme pour l'avenir, mais une expérience de pensée utile consiste à explorer comment la sécurité de ce système à trois joueurs se dégrade pour atteindre une sécurité du niveau de Bitcoin lorsque le PaySplit atteind des valeurs extrêmes.

Saito

Plus d'informations sur la sécurité du réseau.

La conception de Saito résout plusieurs problèmes de longue date. Les attaques par thésaurisation (accumulation) sont minimisées car les nœuds participant au routage des transactions maximisent leurs revenus en trouvant le chemin de routage le plus efficace dans le réseau. La concurrence encourage le partage des frais plutôt que leur thésaurisation. La disponibilité des informations de routage par blocs, permet également aux participants de vérifier que leurs pairs propagent adéquatement leurs transactions au lieu de les accumuler. Parce que l'ajout de rebonds à tout chemin de routage réduit nécessairement la rentabilité du routage pour chaque nœud du chemin, les attaques de type sybil sont également éliminées. Les blocs fournissent les informations nécessaires pour que les participants identifient et éliminent les sybilles dans leurs réseaux pair-à-pair. De plus, l'extensibilité du réseau assure qu'ils les élimineront : les nœuds les plus faibles qui se permettent d'être victimes de sybilles se verront, avec le temps, chassés du réseau par la pression de la concurrence. Le réseau de routage sert également de mécanisme défensif unique. Les nœuds de routage de Saito peuvent augmenter le coût d'une attaque en temps réel en refusant d'acheminer les transactions vers les attaquants, forçant ainsi l'attaquant à se reposer sur son propre portefeuille pour financer la production de blocs. Ce mécanisme défend également Saito contre des attaques subtiles comme la monétisation des flux de transactions et le routage à accès fermé (centralisé).

Comme dernière observation, nous notons que le trilemme de l'extensibilité, souvent présenté comme loi fondamentale de la blockchain, n'existe pas dans la conception du consensus Saito. Il existe de nombreuses configurations évidentes du réseau dans lesquelles rediriger les frais des mineurs vers les nœuds de routage peut simultanément augmenter le débit, la décentralisation et la sécurité du réseau.

Saito

En bref.

Les problèmes fondamentaux qui affectent l'extensibilité des blockchains sont d'ordre économique. Saito résout ces problèmes, nous permettant de construire une blockchain massivement extensible qui atteint des débits de données de niveau mondial en s'assurant que les paiements financent les nœuds qui dépensent de l'argent pour l'infrastructure du réseau.

Ceux qui se penchent sur les détails techniques du réseau Saito y trouveront au moins sept innovations majeures de la technologie blockchain : la rediffusion automatique des transactions, le "burn fee" (brûlage des frais), le système de ticket d'or, le PaySplit et le PowSplit, les tickets d'or sur N-block, un mécanisme de vote multipartite sécurisé et la chaîne de signatures cryptographiques qui permet à la blockchain d'identifier et de récompenser les nœuds productifs du réseau de routage.

Ces techniques ont fait l'objet d'une protection par brevet et nous sommes heureux d'être contactés par d'autres projets blockchain qui cherchent à intégrer une ou plusieurs de ces méthodes dans leurs propres réseaux. Nous encourageons également les lecteurs à visiter notre site web (saito.io) qui est une vitrine fonctionelle du consensus avec ses applications décentralisées *on-chain* (Arcade, portefeuille, forum, messagerie instantanée, ...).

 **SaitoTech**

 **Saito**

 **t.me/SaitoIO**

 **SaitoOfficial**

 **SaitoOfficial**

 **Saito**