الورقة البيضاء للبيتكوين

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Translated in Arabic from bitcoin.org/bitcoin.pdf by Ahmed Gad

قام بترجمة الورقة البحثية إلي العربية م/ أحمد جاد معوض Gadzoghby@gmail.com

البيتكوين : نظام دفع إلكتروني قائم علي تكنولوجيا (النظير الي النظير) .

نبذة مختصرة :

- نظام ونسخة أصيلة للمدفوعات الإلكترونية معتمدة علي تكنولوجيا (النظير إلي النظير) ستسمح لطرف ما أن يرسل ويستقبل المال من وإلى طرف اخر مباشرة عبر شبكة الانترنت دون الرجوع الي أي مؤسسة مالية وسيطة (مثل البنوك أو شركات بطاقات الائتمان).

- تكنولوجيا التوقيع الإلكتروني جزء أصيل من النظام لكن الفائدة الرئيسة للنظام المقترح تفقد قيمتها اذا لبثنا في احتياج لطرف ثالث وسيط لمنع مشكلة الإنفاق المتكرر .

- مقترحنا لحل مشكلة الانفاق المتكرر هو استخدام شبكة من الحواسيب الإلكترونية المتصلة بشبكة الانترنت لعمل شبكة نظائر .

- تقوم هذه الشبكة من الحواسيب بدمج المعاملات التجارية المحددة بوقت إرسالها في سلسلة ضخمة معتمدة علي كل من تكنولوجيا الترميز وخوارزم بذل الجهد .

- ثم يقوم النظام بعمل سجل ثابت للمعاملات التجارية غير قابل للتعديل إلا عن طريق القدرة علي إعادة كل المعاملات السابقة التي تمت عن طريق تكنولوجيا بذل الجهد .

- كلما زاد طول سلسلة المعاملات كان هذا دليلا ليس فقط علي صحة المعاملات التجارية التي شهدتها بل أيضا علي أن هذه السلسة أتت من (نقطة التجميع) الكبري لمجموع القوة الحسابية لأجهزة الكمبيوتر التي نفذتها .

- طالما أن غالبية أجهزة الكمبيوتر وقوتها الحسابية المشاركة في سلسلة المعاملات كان هدفهم هو توثيق هذ المعاملات وليس مهاجمة السلسلة نفسها فإنهم سيتفوقون علي الأقلية التي ستهاجم الشبكة بغرض السرقة.

- شبكة الأجهزة نفسها تتميز بالبساطة في تركيبها ، يتم بث المعاملات بناء علي أفضل (بذل جهد) موجود ، والأجهزة المجتمعة في (نقطة تجميع) واحدة تستطيع ترك الشبكة أو الرجوع إليها متي شاءت ، وذلك بقبولها أطول سلسلة من (بذل جهد) في حالة رجوعها إلي الشبكة .

١- مقدمة :

تعتمد التجارة الإلكترونية في معظم الأحيان بشكل حصري علي المؤسسات المالية للعمل كوسيط موثق لتنفيذ المعاملات المالية بين البائع والمشتري في المدفوعات الالكترونية .

بينما يعمل النظام المؤسسي بشكل جيد في معظم المعاملات إلا أن معاناته تكمن في الضعف المتأصل في النموذج المعتمد بشكل كامل علي وجود وسيط في المعاملات .

لذا فإن المعاملات الموثقة بصورة نهائية بدون مرتجعات غير ممكنة في النظام المؤسسي ، نظرا لأن المؤسسات لا تستطيع ان تتجنب التدخل لحل النزاعات الناتجة عن الاحتيال في نظام المدفوعات أوالإستخدام غير الشرعي لبطاقات الإئتمان مثلا.

ونتيجة لهذا التدخل تزداد قيمة الرسوم المدفوعة للوسيط ويزداد معها الحجم اللازم لحفظ بيانات العملاء ويقطع الطريق علي إمكانية إرسال مدفوعات عادية يومية بقيمة صغيرة ، وإضافة لذك فهناك خسارة أكثر فداحة نتيجة فقدان هذه المؤسسات القدرة علي تقديم مدفوعات ليس بها مرتجعات كخدمات للعملاء الذين لا يحتاجون مثل هذا النوع من التدخل .

مع إزدياد الحاجة لمعاملات صحيحة غير ناتجة عن احتيال تزداد الحاجة الي طرف وسيط لحل النزاعات ، لذا فإن المؤسسات المالية المطلوب منها التدخل تزعج عملاءها بكثرة المعلومات التي بجب ان يقدموها عن أنفسهم وفي معظم الأحيان تكون هذه المعلومات اكثر مما ينبغي وذلك لضمان صحة معاملاتهم ، وعلي الرغم من ذلك فان هناك نسبة من المعاملات التي تحدث نتيجة احتيال لا تستطيع هذه المؤسسات استرجاعها وتعتبر في حيز المفقودة إلي الأبد.

بالطبع أنت تستطيع أن تتجنب الإحتيال أو حتي الإضطرار إلي دفع رسوم إضافية إذا استخدمت عملات نقدية ورقية لكنه حتي الأن لا يوجد نظام يضمن لك صحة معاملاتك المالية عبر قناة اتصال (عن بعد) دون وجود طرف ثالث ضامن ، مانقدمه الآن هو نظام دفع إلكتروني لا يعتمد علي الثقة ولكن علي تشفير المعلومات المنقولة . مما يسمح لطرفين بارسال معاملاتهم مباشرة إالي بعضهم البعض دون الحاجة الي طرف ثالث ضامن .

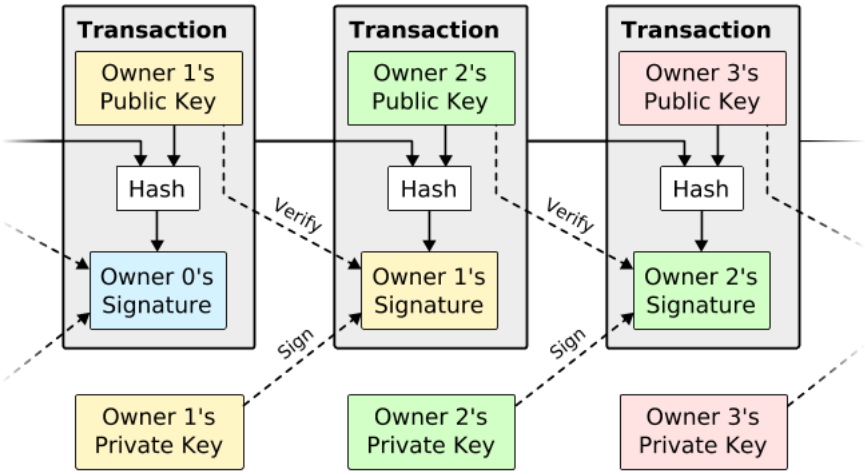
وهذه المعاملات المعتمدة علي القوة الحسابية لأجهزة الكمبيوتر والمعتمدة علي نظام تشفير يستحيل عمليا كسره هي معاملات غير مرتجعة لحماية البائعين من عمليات الإحتيال ، ويمكن بسهولة ضمان حق المشترين عن طريق تطبيق أليات الضمان الروتينة .

في هذا البحث نقدم حلا لمشكلة الإنفاق المتكرر عن طريق نظام يعتمد علي تكنولوجيا النظائر مبني علي (دفتر أستاذ) إلكتروني موزع علي أجهزة الحاسب المشاركة في النظام مدمج به المعاملات مرتبة بناءً علي وقت إرسالها .

وكما ذكرنا سابقا يظل النظام محمي تماما طالما كانت غالبية أجهزة الكمبيوتر وقوتها الحسابية المشاركة في سلسلة المعاملات مكونة سلسلة أطول وقوة حسابية أعلي من سلسلة المعاملات التي ممكن ان تنتج عن هجوم الكتروني علي الشبكة.

٢- المعاملات (Transactions) :

رؤيتنا للعملة الإلكترونية تتلخص في كونها سلسلة من التوقيعات الإلكترونية ، يستطيع أي مالك حالي للعملة تحويل عملاته إلي الطرف التالي (المالك الجديد للعملة) عن طريق التوقيع الإلكتروني ، وهذا التوقيع يتم عن طريق دمج ترميز المعاملة السابقة مع مفتاح رقمي يسمي (المفتاح العام) يمثل عنوان المالك التالي للعملة وإضافة هذا التوقيع في نهاية العملة (بما ان العملة تتكون من سلسلة من التوقيعات كما ذكرنا) ، ويستطيع المستفيد التحقق من التوقيعات الإلكترونية للتحقق من سلسلة المالكين .



تكمن المشكلة الحقيقية بالنسبة للمستفيد في كونه لا يستطيع التحقق من أن أحد المالكين لم يقم بتكرار إنفاق العملة ، الحل العام المستخدم في النظام المؤسسي هو وجود طرف ضامن (سلطة مركزية أو مصلحة صك العملة) للتحقق من كل معاملة لمنع الإنفاق المتكرر للعملة .

وأي عملة موجودة في السوق لم تصدر مباشرة من مصلحة صك العملة لا يعتد بها ، المشكلة بالطبع في هذا النظام المؤسسي هي أن النظام المالي بالكامل يعتمد كليا علي من يدير مصلحة صك العملة نظرا لأن كل عملة يجب أن تمر من خلالهم فقط .

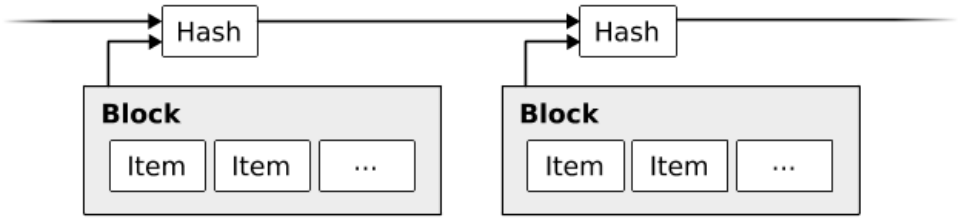
المطلوب هو إيجاد طريقة للمستفيد ليكون علي دراية تامة بأن من سبقوه في امتلاك العملة لم يقوموا بإنفاقها اكثر من مرة قبل وصولوها إليه .

لهذا الغرض تعتبر المعاملة الأولي في النظام هي الأكثر أهمية للتركيز عليها ، وبهذا لا يهمنا المحاولات التالية للإنفاق المتكرر ، والطريقة الوحيدة للتأكد من فقدان معاملة أو صحتها هي الدراية الكاملة بكل المعاملات التي سبقت المعاملة التي بين يديك الان (معرفة تامة لتاريخ المعاملات منذ أول عملة إلي المعاملة التي وصلتك) .

في النظام المؤسسي جهة صك العملة هي التي تكون علي علم بكل المعاملات وترتيب صدورها ، ولتحقيق هذ في النظام المقترح بدون وجود طرف ضامن ثالث يجب علي كل المعاملات أن تكون علنا (معروفة ومتاحة للجميع) .وبالطبع يجب علي هذا النظام أن يسمح لكل المشتركين أن يتفقوا علي نسخة واحدة موثقة لتاريخ جميع المعاملات ، لذا فإن المستفيد يحتاج إلي إثبات من أغلبية المشتركين في الشبكة أنه في وقت إستلامه للمعاملة كانت هذه المعاملة أصيلة وفريدة ولم تتكرر.

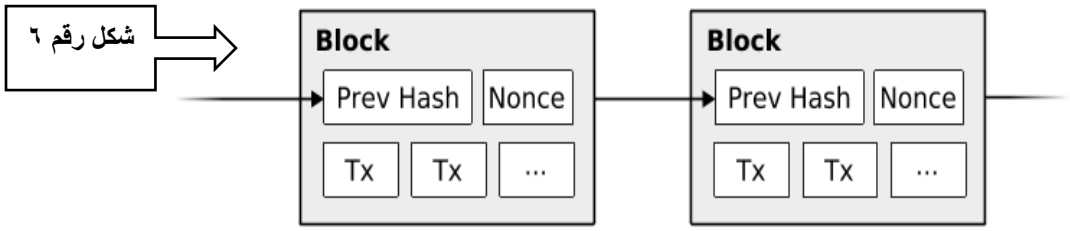
٣ - حاسوب ضبط الوقت (Timestamp Server):

النظام المقترح قائم علي حاسوب مسئول عن ضبط وقت المعاملات ، ويعمل هذا الحاسوب علي أخذ ترميز فريد لكتلة من العناصر واخذ وقت حدوثها وتدونيه بدقة ثم يقوم بنشرهذا الترميز علي كامل نطاق الشبكة ، كما هو الحال في الصحف او المجموعات البريدية الإخبارية ، وبهذا يكون الختم الزمني المرافق للمعاملة هو دليل علي حدوث هذه المعاملة في توقيتها السليم المتاح للجميع في الشبكة ، وتتكون سلسلة المعاملات السليمة عن طريق أن كل ترميز لمعاملة يحتوي علي ختم زمني للمعاملة السابقة ، وبهذا يقوم كل ترميز جديد علي تعزيز الترميز الذي قبله .



٤ - خواريزم بذل الجهد (Proof-of-Work):

لإمكانية تطبيق حاسوب ضبط الوقت بنظام موزع (ليس به نقطة مركزية) قائم علي تكنولوجيا النظائر فإننا سنكون بحاجة إلي نظام بذل الجهد مشابه لنظام Adam Back's Hash (ادم باك مخترع وعالم تشفير شهير) كما هو موضح بالشكل رقم ٦، عوضا عن نظام الصحف أو المجموعات البريدية المذكورة سابقا.



خواريزم (بذل الجهد) يتضمن البحث عن قيمة الترميز (كما هو مستخدم في خوارزمية SHA-256) -الترميز يبدأ بعدد من الأصفار ، متوسط العمل الازم لفك الترميز يتانسب طرديا (بعلاقة أسية) مع عدد الأصفار ويتم التحقق منه عن طريق فك ترميز عملية واحدة .

بالنسبة لشبكة المعاملات المختومة بحاسوب ضبط الوقت فإنه يتم بداخلها زرع خواريزم بذل الجهد بزيادة قيمة معلومة (تسمي nonce) في كتلة المعاملات تستخدم لإثبات صحة الأصفار في الترميز المطلوب وبالتالي صحة المعاملات داخل الكتلة ، بمجرد إستهلاك قوة المعالجة الحسابية في معرفة ترميز كتلة المعاملات فإنك لا تستطيع تغيير قيمة أي شئ داخل الكتلة بدون إعادة استخدام نفس القوة الحسابية مرة أخري وبما ان الكتل مرتبطة ببعضها كما شرحنا سابقا فانه لتغيير قيمة أي كتلة معاملات يجب عليك تغيير قيمة كل الكتل السابقة لهذه الكتلة مما يعد استحالة .

يستخدم خواريزم بذل الجهد أيضا في حل مشكلة كيفية تحديد رأي الأغلبية السائد لتحديد أطول سلسلة معاملات ، لأنه اذا كان رأي الأغلبية يعتمد علي فرضية كل من يملك عنوان إلكتروني (IP) يملك حق التصويت بصوت واحد لصار رأي الأغلبية محكوماً بمن يستطيع جمع أكبر عدد من العناوين (IPs) الذي بدوره قد يؤدي إلي تخريب قرار الاغلبية .

بينما خواريزم بذل الجهد يخصص صوتاً واحداً لكل قوة معالج حسابي ورأي الأغلبية تحدده أطول سلسلة معاملات لأقوي مجموع لقوى المعالجات الحسابية وبالتالي أقوي بذل جهد مبذول بواسطتها .

وإذا تم التحكم في أقوي سلسلة للمعالجات الحسابية عن طريق (عُقد صادقة) فإنها ستنمو أسرع وتتغلب علي كل السلاسل المنافسة المهاجمة بغرض السرقة أو التخريب.

لكي يستطيع أي أحد إختراق سلسلة المعاملات وتعديل أي كتلة يجب عليه أن يعيد انتاج نفس قوة المعالجات التي تم بها توثيق هذه الكتلة وكل الكتل التي تليها ، ثم بعد ذلك عليه أن يتفوق علي سلسلة (العقد الصادقة) ، سنوضح لاحقا أن إحتمالية قدرة مهاجم أو مخرب علي تحقيق هذا تتلاشي بطريقة أسية مع ازدياد حجم سلسلة (العقد الصادقة) بإضافة كتل معاملات جديدة.

لتعويض الأثر الناتج من القوة الحسابية لأجهزة الكمبيوتر وأيضا تعويض النقص الناتج في العقد نتيجة نقص الإهتمام مع مرور الوقت ، يتم تحديد الصعوبة في خواريزم بذل الجهد يستهدف متوسط عدد ثابت من الكتل في الساعة ، بمعني أنه إذا زادت السرعة في إنشاء الكتل تزداد الصعوبة والعكس صحيح .

٥ - الشبكة (Network):

خطوات عمل الشبكة تكون علي النسق التالي :

١- يتم بث المعاملات الجديدة تباعا إلي كل العقد الصادقة.
٢- تعمل كل عقدة علي تجميع المعاملات الجديدة داخل كتل المعاملات.
٣- تعمل كل عقدة علي إيجاد حل لصعوبة خواريزم (بذل الجهد) لكل كتلة.
٤- عندما تنجح العقدة في حل الصعوبة تقوم ببث كتل المعاملات إلي باقي العقد في الشبكة.
٥- تقوم العقد بقبول الكتلة الجديدة فقط إذا كانت كل المعاملات التي بداخلها صحيحة ولم يتم إنفاقها من قبل.
٦- وتعرب العقد عن قبولها للكتلة الجديدة بالبدء في حل صعوبة الكتلة التالية في سلسلة المعاملات باستخدام ترميز الكتلة المقبولة حاليا كترميز سابق مدمج في الكتلة التالية .

-العقد دائما ستعتبر سلسلة المعاملات الأطول هي الأصح وستستمر في زيادتها ، في حالة تلقي عقدتان لكتلتين مختلفتين من المعاملات في نفس الوقت فقد يحدث ان تجذب العقدة هذه الكتلة أو تلك المختلفة عنها وفي هذه الحالة ستعمل العقدة علي الكتلة التي تلقتها أولا لكنها ستحتفظ بالكتلة الأخري في فرع أخر من المعاملات في حالة أن أصبح هذا الفرع هو السلسة الأطول ، بعدها سيتم كسر حالة التعادل هذه عندما يتم حل خواريزم بذل الجهد التالي الذي سيحدد من هي السلسة الاطول وفي هذه الحالة سيتم انتقال الكتلة الاخري إلي الفرع الأطول .

بث المعاملات الجديدة لا يتطلب بالضرورة الوصول إلي جميع العقد طالما انها وصلت الي العديد من العقد فستصل إلي أقرب كتلة معاملات في أقرب وقت ، إذا فقدت عقدة ما كتلة معاملات فإنها ستتدارك هذا بمجرد الوصول الي الكتلة التالية فستعرف ان هناك كتلة مفقودة وتعيد تنظيمها .

٦- الحافز (Incentive):

منطقيا ستكون أول معاملة في أي كتلة معاملات ذات طبيعة خاصة لأنها ستنشئ عملة جديدة تكون ملكا لمنشئ الكتلة ، وهذا بحد ذاته حافز كبير للعقد لدعم الشبكة ، ويقوم بإيجاد طريقة فعالة لتوزيع العملات للتداول .

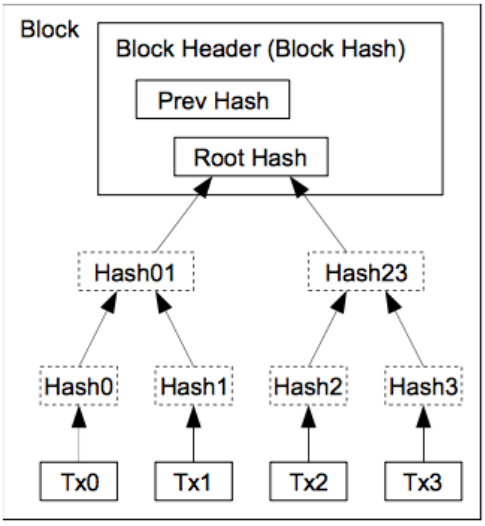
وحيث أنه لاتوجد سلطة مركزية لإصدار العملات فإن الإضافة الثابتة لكمية من العملات وتداولها سيكون مماثلا لعمال المناجم الذين ينفقون الوقت والمجهود لإضافة قطع ذهبية جديدة الي السوق ، وفي حالتنا بالطبع الوقت والجهد المبذول هو وقت وجهد الكهرباء المبذولة و القوة الحسابية لأجهزة الكمبيوتر المكونة للشبكة ، ونستطيع بالطبع زيادة الحافز بإضافة رسوم .

لذا إن كانت قيمة مخرجات معاملة أقل من قيمة مدخلاتها فإن الفرق يكون قيمة رسوم تضاف كحافز لتعدين الكتلة التي تحوي المعاملة ، وبمجرد نفاذ عدد العملات المحددة مسبقا (٢١ مليون عملة) يكون الحافز معتمداً بالكامل علي رسوم التعدين وبالتالي خالي تماما من التضخم نظرا لثبات عدد العملات مع زيادة الطلب عليها .

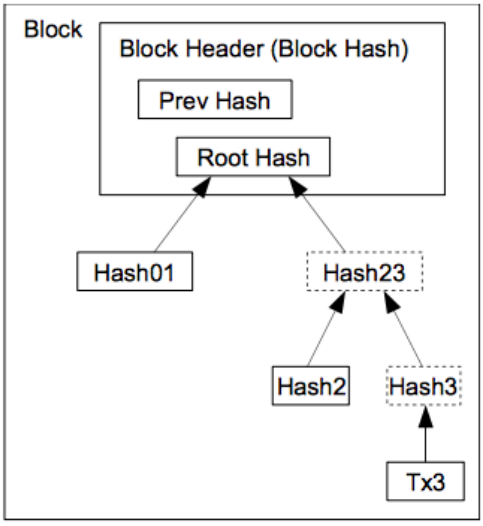
سنجد أيضاً ان الحافز قد يساعد علي دعم العقد أن تكون صحيحة وصادقة ، لأنه اذا حاول مهاجم جشع أن يتفوق بقوة المعالجات الحسابية لأجهزته لمهاجمة الشبكة فإنه في الحقيقة سيجد انه اذا إلتزم بالقواعد التي تحافظ علي نقاء العقد واستخدم قوة أجهزته في إيجاد عملات جديدة فانه سيحقق ربحا أكثر من الذي سيجنيه من مهاجمة الشبكة و رد مدفوعاته إليه مرة اخري عن طريق الإحتيال وأنه بإضعاف الشبكة قد يهدد ثروته التي جناها .

٧ - الحفاظ علي مساحة وحدة التخزين (Reclaiming Disk Space):

- طالما أن المعاملة الأخيرة سبقها القدر الكافي من الكتل السليمة فإن المعاملة التي سبقتها تصبح غير ذات جدوي ونستطيع توفير مساحتها التخزينية في القرص الصلب ، ولتسهيل ذلك دون كسر ترميز الكتل فإنه سيتم ترميز المعاملات عن طريق خوارزم (Merkel tree) الذي يسمح بدمج ترميز المعاملات ثنائيا حتي نصل الي ترميز الجذر فقط للكتلة ويتم ضغط الكتل القديمة إلي مجموعة من الأفرع ولا يتم تخزين الترميزات الداخلية .



Transactions Hashed in a Merkle Tree

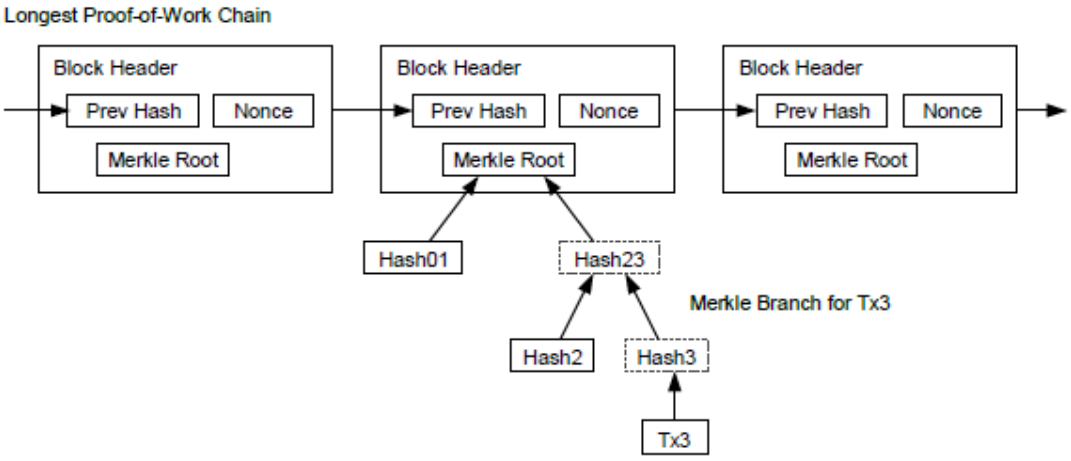


After Pruning Tx0-2 from the Block

مساحة عنوان أي كتلة ستكون تقريبا ٨٠ بايت ، وبفرض انه سيتم إنشاء كتلة معاملات كل عشر دقائق فإن الحجم الازم لتحخزين الكتل في عام سيكون ٨٠*٦*٢٤*٣٦٥= ٤ ميجا بايت . وبحساب جهاز كمبيوتر ذو ذاكرة عشوائية تساوي ٢ جيجا بايت للعام ٢٠٠٨ وبتطبيق قانون موور الذي يتنبأ بزيادة حجم الذاكرة والقوة الحسابية للأجهزة بقيمة تقريبية ١,٢ جيجا بايت للعام فإننا سنجد أن المساحة التخزينية للكتل لا تمثل أي مشكلة علي الإطلاق حتي لو تم تخزين عناوين الكتل في الذاكرة .

٨ - التحقق من المدفوعات (Simplified Payment Verification):

من الممكن التحقق من المدفوعات دون الإضطرار الي تفعيل عقدة كاملة من الشبكة . في الحقيقة كل مايحتاجه المستخدم هو الإحتفاظ بنسخة من عناوين كتل المعاملات لأطول سلسلة خوارزم بذل الجهد ، والذي يمكن تحقيقه من خلال الإستعلام عن عقد الشبكة حتى يطمئن المستخدم بانه موجود علي أطول سلسلة ، ثم الحصول على فرع شجرة Merkelالذي يربط المعاملة بالكتلة المختومة زمنيا المربوطة بحاسوب ضبط الوقت ، بالطبع لا يستطيع المستخدم تتبع المعاملة بنفسه لكنه بربط المعاملة بمكان محدد في السلسلة لكي يستطيع أن يري أن عقدة صادقة من الشبكة قد قامت بقبول معاملته . وقامت كتلة معاملات بإضافتها إليها بعد التحقق منها.



وبهذا سيكون التحقق من صحة المعاملات سليما وموثوقا طالما كانت العقد الصادقة هي التي تتحكم بالشبكة ، ولكنها ستكون عرضة للتلاعب اذا إستطاع المهاجم ان يتفوق علي القوة الحسابية لمجموع قوة الشبكة .

وبينما تستطيع عقد الشبكة التحقق من المعاملات بنفسها إلا أن وجود مهاجم بقوة حسابية أعلي من مجموع االقوة الحسابية للشبكة يمكن أن يتسبب في التلاعب في الطريقة المستخدمة للمدفوعات وأن يقوم بتلفيق معاملات .

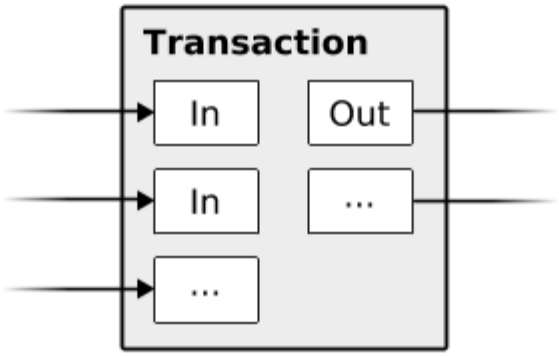
وأحد الحلول لمثل هذا السيناريو هو قبول تحذيرات من العقد في حالة رصدها لأي كتل غير سليمة ، مما يدفع برنامج المستخدم إلى تنزيل الكتلة الكاملة والتحقق من المعاملات التي لا تتسق معها لكشف التناقض.

قد يرغب بعض أصحاب الأعمال الذين يتلقون مدفوعات كثيرة في تكوين عقد خاصة بهم لزيادة الامان وتسريع عملية التحقق من المعاملات .

٩ - دمج وتقسيم قيمة المعاملات (Combining and Splitting Value) :

على الرغم أنه من الممكن التعامل مع العملات بشكل فردي ، إلا أن هذا سيكون غير عملي إجراء عملية تحويل كاملة لكل سنت في المعاملات . لذا لكي نسمح بعمليات دمج وتقسيم قيمة المعاملات سيكون هناك عدة مدخلات ومخرجات .

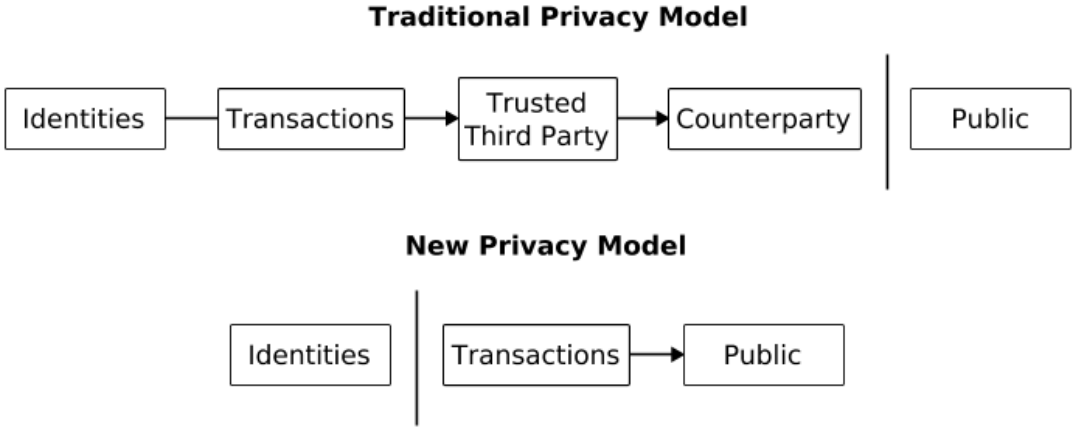
وفي الغالب سيكون هناك إما مدخل وحيد بقيمة ضخمة مسبقة أو عدة مدخلات ناتجة عن مجموع قيم أصغر ، وسيكون هناك غالباً مخرجين فقط أحدهما للمدفوعات والاخر للباقي إن وجد والباقي بالطبع سيعود مرة أخري للمرسِل .



ويجب الإشارة إلي أن جميع التحويلات مرتبطة بشكل تام ، وذلك لأن أي معاملة مرتبطة بعدة معاملات التي بدورها معتمدة علي عدة معاملات أخري ، وهذا لايمثل أي مشكلة علي الإطلاق لأننا لن نكون أبدا في حاجة لإمتلاك نسخة كاملة مستقلة من تاريخ كل المعاملات .

١٠- الخصوصية (Privacy):

تعتمد الخصوصية في النظام التقليدي (نظام البنوك) علي الحد من الوصول إلي المعلومات عن طريق الطرف الثالث أو الأطراف المعنية الأخري ، وبالطبع هذه الطريقة غير مجدية في نظامنا المقترح القائم علي علانية جميع المعاملات ، ولكن الخصوصية والسرية ممكن أن تستمر بكسر سيل المعلومات الخاصة عن طريق آخر وهو الإبقاء علي المفاتيح العامة مجهولة الهوية ، وبهذا يستطيع العامة أن يروا أن هناك شخص ما يرسل كمية محددة من المال إلي شخص آخر ولكن لاأحد يستطيع معرفة أي تفاصيل عن ماهية هؤلاء الأشخاص أو معرفة أي رابط بين تلك المعاملات وهذا الشخص ، وهو مستوي أمان مماثل للمعروض في بورصات الأسهم العالمية ، حيث وقت وحجم سهم معين (معاملة) معروف للعامة لكن غير معرفة ماهية الأشخاص أو الهيئات المشتركة في تداول هذا السهم .



وكحائط ناري إضافي يجب إستخدام زوج مختلف من المفاتيح(التوقيع الإلكتروني) كل مرة يقوم المستخدم فيها بإرسال معاملة لمنع إرتباطهم بمالك معين ، ولكن سيظل هناك بعض الإرتباطات التي لا يمكن تجنبها وهي التي تحدث نتيجة المدخلات المتعددة التي من الضروري إظهارها حتي يتسني لنا معرفة أن لهم نفس المالك . ويكمن الخطر في أنه في حالة الكشف عن مالك المفتاح ، قد يكشف الارتباط عن معاملات أخري تخص نفس المالك .

١١- العمليات الحسابية (Calculations) :

لنفترض السيناريو التالي :

هناك مهاجم يحاول أن يخلق سلسلة مغايرة عن السلسة ذات العقد الصادقة بقوة حسابية أقوي منها ، حتي لو نجح في مبتغاه فإنه لن يستطيع جعل النطام يحدث تغييرات إعتباطية مثل خلق أموال من الفراغ (لا أصل لها) أو أخذ أموال لا تنتمي إليه في المقام الاول ، أي أن المهاجم لا يملك إلا تغيير معاملة واحدة فقط من معاملاته وهي إستعادة اموال كان قد سبق وأن أنفقها لأن العقد لن تقبل معاملات غير صحيحة كوسيلة دفع وبالتالي فإن العقد الصادقة لن تقبل كتلة متحتوي علي مثل هذه المعاملات .

يتميز السباق بين العقد الصادقة وغيرها بأنه سباق عشوائي ذو حدين (Binomial Random Walk) ، لذا فإن الحدث الناجح هو زيادة العقد الصادقة بكتلة واحدة مما يجعلها تسبق بقيمة +١ ، والحدث الفاشل هو زيادة سلسلة المهاجم بكتلة واحدة مما يجعلها تنقص بقيمة -١ ، وبالتالي فإن إحتمالية قدرة المهاجم علي تغطية عجزه مماثل لمعضلة إفلاس المقامر (Gambler’s ruin problem)

والتي يمكن تلخيصها كالتالي :

افترض أن هناك مقامر بكمية غير محدودة من المال بدأ بالخسارة ومضطر للعب عدد لا نهائي من المحاولات حتي يعوض خسارته ، نستطيع الان أن نحسب احتمالية تعويض خسارته المقابلة لاحتمالية المهاجم المتراجع أن يصل إلي السلسلة ذات العقد الصادقة وهي كالتالي :

P = إحتمال أن عقدة صادقة تجد الكتلة التالية.
q = إحتمال أن يجد المهاجم الكتلة التالية.
q_z = إحتمال لحاق المهاجم إذا كان متخلفا بمقدار z كتلة.

اذا :

$$q_z = \begin{cases} 1 & if\ p \le q \\ (q/p)^z & if\ p > q \end{cases}$$

مع فرضية أن (p > q) فإن إحتمالية أن يستطيع المهاجم اللحاق بسلسلة العقد الصادقة تنهار بشكل أسي كلما زاد عدد الكتل التي يجب علي المهاجم تخليقها ، ومع كون الإحتمالات ضده مالم يساعده حظه باندفاع مبكر في الشبكة فإن فرصِه في اللحاق بشبكة العقد الصادقة تتلاشي تقريبا مع إستمرار تراجعه .

وبما أننا الان علي علم بالوقت الذي يحتاجه المُرسل إليه لكي يتأكد من أن الراسل لايستطيع تغيير المعاملة مرة أخري ، ومع فرضية أن الراسل مهاجم للشبكة الذي يحاول إقناع المتلقي لفترة ما أنه قام بإرسال معاملة سليمة ثم يقوم المهاجم بتحويل هذا المال مرة أخري إلي نفسه ، بالطبع سيتم تنبيه المُرسل إليه بحدوث هذا لكن المهاجم سيأمل ان يكون الوقت قد تأخر لفعل أي شئ تجاه المعاملة المردودة .

سيقوم المُرسل إليه بتوليد زوج جديد من المفاتيح ويقوم بعملية التوقيع الرقمية ثم يبعث بالمفتاح العام (التوقيع الإلكتروني) إلي الراسل وهذا سيقوم بمنع المهاجم من أن يقوم بتجهيز شبكة مسبقة من كتل المعاملات تسبق سلسلة العقد الصادقة التي يعمل عليها بصفة مستمرة حتي يحالفه الحظ ليسبق الشبكة بقدر كاف ، ثم تنفيذ المعاملة في نفس هذا التوقيت ، وبمجرد إرسال المعاملة سيقوم المهاجم سرا بالعمل علي شبكة موازية للعقد الصادقة تحتوي علي نسخة مغايرة للمعاملة المذكورة .

في هذا الوقت سيكون المُرسل إليه في إنتظار المعاملة حتي تضاف إلي كتلة معاملات ثم ربطها بعدد Z من الكتل بعدها ، ومع عدم علمه إلي أي مدي قام المهاجم بتطوير عمله وبفرض ان كتل المعاملات استهلكت متوسط الوقت المحدد للكتلة ، فإن مدي تقدم المهاجم في عمله سيندرج تحت توزيع بويسن (poisson Distribution) بقيمة متوقعة كالتالي :

$$\lambda = z\frac{q}{p}$$

ولحساب إحتمالية ان يستطيع المهاجم أن يسبق الشبكة في تلك اللحظة سنقوم بحساب كثافة توزيع بويسن لكل مقدار من تقدمه مضروبة في إحتمالية تمكنه باللحاق بالشبكة في تلك اللحظة كالتالي :

$$\sum_{k=0}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \cdot \begin{cases} (q/p)^{(z-k)} & if\ k \le z \\ 1 & if\ k > z \end{cases}$$

وبإعادة ترتيب المعادلة لتجنب جمع اللانهاية :

$$1 - \sum_{k=0}^{z} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - (q/p)^{(z-k)}\right)$$

وبالتحويل إلي كود بلغة C :

```
#include <math.h>
double AttackerSuccessProbability(double q, int z)
{
   double p = 1.0 - q;
   double lambda = z * (q / p);
   double sum = 1.0;
   int i, k;
   for (k = 0; k <= z; k++)
   {
      double poisson = exp(-lambda);
      for (i = 1; i <= k; i++)
      poisson *= lambda / i;
       sum -= poisson * (1 - pow(q / p, z - k));
    }
    return sum ;
}
```

وبالقيام بتنفيذ الكود وقراءة بعض النتائج (المخرجات) سنجد أن الاحتمالية تنهار بقيمة أسية بالنسبة إلي Z :

q=0.1

z=0 P=1.0000000
z=1 P=0.2045873
z=2 P=0.0509779
z=3 P=0.0131722
z=4 P=0.0034552
z=5 P=0.0009137
z=6 P=0.0002428
z=7 P=0.0000647
z=8 P=0.0000173
z=9 P=0.0000046
z=10 P=0.0000012

q=0.3

z=0 P=1.0000000
z=5 P=0.1773523
z=10 P=0.0416605
z=15 P=0.0101008
z=20 P=0.0024804
z=25 P=0.0006132
z=30 P=0.0001522
z=35 P=0.0000379
z=40 P=0.0000095
z=45 P=0.0000024
z=50 P=0.0000006

وبالنطر إلي المخرجات مع قيمة P أقل من 0.1% :

P < 0.001

q=0.10 z=5
q=0.15 z=8
q=0.20 z=11
q=0.25 z=15
q=0.30 z=24
q=0.35 z=41
q=0.40 z=89
q=0.45 z=340

١٢- ملخص ختامي:

قمنا بعرض نظام لمعاملات مالية إلكترونية لا يعتمد علي الثقة ، وبدأنا بالإطارالمعتاد للعملات المعتمدة علي التوقيع الإلكتروني ، الذي يتيح تحكم كامل في الملكية ، لكن هذا النظام لا يعتد به بدون طريقة فعالة لمنع الإنفاق المتكرر للعملة .

ولحل هذه المشكلة إقترحنا نظاما معتمداً علي تكنولجيا النظائر مدمج مع خواريزم بذل الجهد مع وجود سجل عام متاح للمعاملات التي تتطور حسابيا بشكل سريع لتصبح مستحيلة عمليا علي مهاجم أن يخترقها أو يقوم بتغييرها إذا كانت العقد الصادقة المكونة لسلسلة المعاملات تمتلك المجموع الاكبر لقوة المعالجات الحسابية .

تتميز الشبكة بقوتها المهولة المتمثلة في بساطة تركيبها وعشوائية مشتركيها ، تعمل العقد مع بعضها جميعا في نفس الوقت مع القليل من التنسيق ، مع عدم الضرورة علي الإطلاق أن تكون هذه العقد معرفة طالما ان الرسائل لا يتم توجيها إلي مكان محدد بعينه ويعتمد توصيلها فقط علي خواريزم أفضل بذل جهد ، تستطيع العقد مغادرة الشبكة والرجوع إليها متي شاءت طالما انه يتم قبول سلسلة بذل الجهد كدليل علي صحة ماحدث أثناء مغادرتها الشبكة .

تقوم العقد بالتصويت معتمدة علي مجموع قوة معالجة أجهزتها الحسابية معربة عن قبولها للكتل الصالحة بإضافتها إلي سلسلة بذل الجهد ومنعها للكتل غير الصالحة من خلال رفضها العمل عليها .وعن طريق أليه التصويت و الإجماع يمكن إضافة أي قواعد إضافية مطلوبة أو حوافز معينة .

المراجع:

1-**W**.Dai,"b-money,"http://www.weidai.com/bmoney.txt",1998.

2 - **H**. Massias, X.S. Avila, and J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirements, ," In 20th Symposium on Information Theory in the Benelux", May 1999 .

3 **S**.Haber, W.S. Stornetta, "How to time-stamp a digital document," In Journal of Cryptology, vol 3 no 2, pages 99-111, 1991.

4 - **D**. Bayer, S. Haber, W.S. Stornetta, "Improving the efficiency and reliability of digital time-stamping," In Sequences II: Methods in Communication, Security and Computer Science, pages 329-334, 1993.

5 - **S**. Haber, W.S. Stornetta, "Secure names for bit-strings," In Proceedings of the 4th ACM Conference on Computer and Communications Security, pages 28-35, April 1997.

6 - **A**. Back, "Hashcash - a denial of service counter-measure," http://www.hashcash.org/papers/hashcash.pdf , 2002.

7 - **R**.C. Merkle, "Protocols for public key cryptosystems," In Proc. 1980 Symposium on Security and Privacy, IEEE Computer Society, pages 122-133, April 1980.

8 - **W**. Feller, "An introduction to probability theory and its applications," 1957.