বিটকয়েন: একটি পিয়ার-টু-পিয়ার বৈদ্যুতিক ক্যাশ সিস্টেম

সাতোশি নাকামোটো
satoshin@gmx.com www.bitcoin.org

সারাংশ: সারাংশ: বৈদ্যুতিক মাধ্যমে নগদ টাকার ক্ষেত্রে বিশুদ্ধ পিয়ার-টু-পিয়ার সংস্করণ লেনদেনের জন্য কোনও আর্থিক প্রতিষ্ঠানের মাধ্যমে না গিয়ে সরাসরি এক পক্ষ থেকে অন্য পক্ষের কাছে অনলাইন পেমেন্ট পাঠানোর সুবিধা করে দেয়। ডিজিটাল স্বাক্ষরগুলি আংশিক সমাধানসূত্র প্রদান করলেও মূল সুবিধাগুলি হারিয়ে যায় যদি দ্বিগুণ-ব্যয় রোধ করার জন্য এর পরেও একটি বিশ্বস্ত তৃতীয় পক্ষের প্রয়োজন হয়। আমরা এক্ষেত্রে একটি পিয়ার-টু-পিয়ার নেটওয়ার্ক ব্যবহার করে দ্বিগুণ-ব্যয়ের সমস্যা রোধ করার জন্য একটি সমাধান হাজির করেছি। নেটওয়ার্ক টাইমস্ট্যাম্প লেনদেনগুলিকে হ্যাশ-ভিত্তিক প্রুফ-অফ-ওয়ার্কের একটি চলমান শৃঙ্খলে হ্যাশ করার মাধ্যমে এমন একটি রেকর্ড গঠন করে যা প্রুফ-অফ-ওয়ার্ককে ফের নতুন করে তৈরি না করা পর্যন্ত বদলানো সম্ভব নয়। দীর্ঘতম শৃঙ্খলটি কেবল প্রত্যক্ষ করা ঘটনাক্রমের প্রমাণ হিসাবে কাজ করে না, তা প্রমাণ করে যে এটি সিপিইউ পাওয়ারের বৃহত্তম পুল থেকে এসেছে। যতক্ষণ পর্যন্ত অধিকাংশ সিপিইউ পাওয়ার সেইসব নোড দ্বারা নিয়ন্ত্রিত হয় যারা নেটওয়ার্ক আক্রমণ করতে সহযোগিতা করছে না, ততক্ষণ পর্যন্ত তারা দীর্ঘতম শৃঙ্খল গঠন করে এবং আক্রমণকারীদের বাঁধা দেয়। এই নেটওয়ার্কের নিজের ন্যূনতম কাঠামো প্রয়োজন হয়। তথ্যগুলি সেরা প্রচেষ্টার ভিত্তিতে সম্প্রচার করা হয়, এবং নোডগুলি ইচ্ছামত নেটওয়ার্ক ছেড়ে যেতে এবং পুনরায় যোগদান করতে পারে। তাদের অনুপস্থিতির সময় যা ঘটেছিল তার প্রমাণ হিসেবে নোডগুলি দীর্ঘতম প্রুফ-অফ-ওয়ার্ককেই মান্যতা দেয়।

১. ভূমিকা

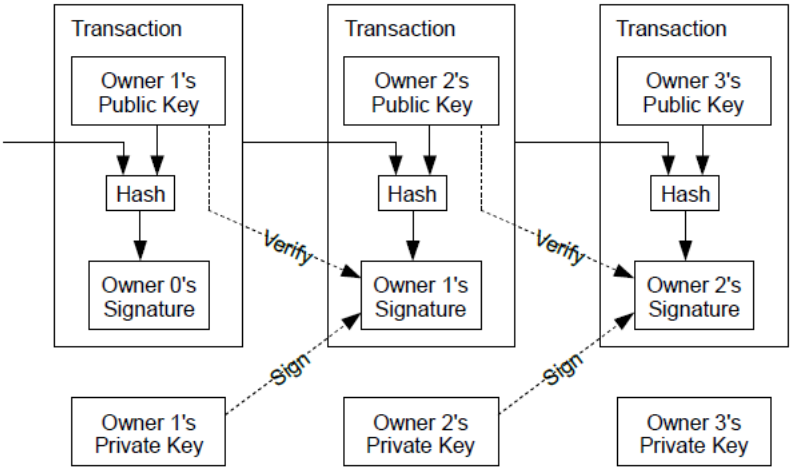
ইন্টারনেটে বাণিজ্যের ক্ষেত্রে বৈদ্যুতিক অর্থপ্রদান প্রক্রিয়া সম্পন্ন করার জন্য বিশ্বস্ত তৃতীয় পক্ষ হিসাবে কাজ করা আর্থিক প্রতিষ্ঠানগুলির উপরে প্রায় একচেটিয়াভাবে নির্ভর করতে হয়েছে। যদিও এই পদ্ধতিটি বেশিরভাগ লেনদেনের জন্য যথেষ্ট ভালভাবেই কাজ করে, কিন্তু এটি এখনও বিশ্বাস ভিত্তিক মডেলের অন্তর্নিহিত দুর্বলতায় ভুগছে। যেহেতু আর্থিক প্রতিষ্ঠানগুলি মধ্যস্থতাকারীদের সঙ্গে বিরোধ এড়াতে পারে না, তাই সম্পূর্ণরূপে অ-প্রত্যাবর্তনযোগ্য লেনদেন সত্যিই সম্ভব নয়। মধ্যস্থতার খরচ লেনদেনের খরচ বাড়ায়, ন্যূনতম ব্যবহারিক লেনদেনের আকার সীমিত করে এবং ছোট খুচরা লেনদেনের সম্ভাবনা কমিয়ে দেয়, এবং অ-প্রত্যাবর্তনযোগ্য পরিষেবাগুলির জন্য অ-প্রত্যাবর্তনযোগ্য পেমেন্ট করার অক্ষমতার জন্যেও বড় মূল্য দিতে হয়। প্রত্যাবর্তনের সম্ভাবনার সঙ্গেই বিশ্বাসের প্রয়োজন বাড়ে। এই দুর্বলতার কারণে ব্যবসায়ীদের অবশ্যই তাদের গ্রাহকদের সম্পর্কে সতর্ক থাকতে হবে, তাদের প্রয়োজনের চেয়ে বেশি তথ্যের জন্য গ্রাহকদের সতর্কও করতে হবে। এ ক্ষেত্রে একটি নির্দিষ্ট শতাংশের প্রতারণা যে হবেই- তা অনিবার্য হিসাবে গ্রহণ করা হয়। এই খরচ এবং অর্থপ্রদানের অনিশ্চয়তাগুলি আসল কয়েন ব্যবহার করে ব্যক্তিগতভাবে এড়ানো যেতে পারে, তবে বিশ্বস্ত পক্ষ ছাড়া যোগাযোগ চ্যানেলের মাধ্যমে অর্থপ্রদান করার কোনও ব্যবস্থা বর্তমানে উপলব্ধ নেই।

এ ক্ষেত্রে যা প্রয়োজন - তা হল বিশ্বাসের পরিবর্তে ক্রিপ্টোগ্রাফিক প্রমাণের উপর ভিত্তি করে একটি বৈদ্যুতিক পেমেন্ট সিস্টেম, যা দুই ইচ্ছুক পক্ষকে কোনও বিশ্বস্ত তৃতীয় পক্ষের প্রয়োজন ছাড়াই একে অপরের সঙ্গে সরাসরি লেনদেন করার অনুমতি দেয়। যে লেনদেনগুলি গণনাগতভাবে ফিরিয়ে আনা অসম্ভব তা বিক্রেতাদের প্রতারণা থেকে রক্ষা করবে এবং ক্রেতাদের সুরক্ষার জন্য রুটিন এসক্রো প্রক্রিয়াগুলি সহজেই প্রয়োগ করা যেতে পারে। এই প্রবন্ধে, আমরা লেনদেনের কালানুক্রমিক ক্রমানুসারের

গণনাগত প্রমাণ তৈরি করতে একটি পিয়ার-টু-পিয়ার ডিস্ট্রিবিউটেড টাইমস্ট্যাম্প সার্ভার ব্যবহার করে দ্বিগুণ-ব্যয় সমস্যার সমাধানের প্রস্তাব করবো। যতক্ষণ না পর্যন্ত সৎ নোডগুলি আক্রমণকারী নোডগুলির যে কোনও সহযোগী গ্রুপের চেয়ে বেশি সিপিইউ শক্তি নিয়ন্ত্রণ করে, ততক্ষণ পর্যন্ত সিস্টেমটি নিরাপদ।

২. লেনদেন

আমরা একটি বৈদ্যুতিক কয়েনকে ডিজিটাল স্বাক্ষরের একটি চেইন হিসাবে নির্দেশিত করি। প্রতিটি মালিক পূর্ববর্তী লেনদেনের একটি হ্যাশ এবং পরবর্তী মালিকের পাবলিক কী-তে ডিজিটালি স্বাক্ষর করেন এবং কয়েনর শেষে এগুলি যোগ করে কয়েনটিকে পরবর্তীতে স্থানান্তর করতে পারেবেন। একজন প্রাপক মালিকানার চেইন যাচাই করার জন্য স্বাক্ষরের পরীক্ষাও করতে পারেন।



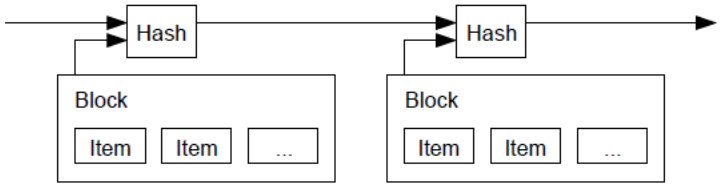
তবে সমস্যাটি, প্রাপক যাচাই করতে পারেন না যে মালিকদের মধ্যে কেউ কয়েনটি দ্বিগুণ খরচ করেছেন কী না। এর একটি সাধারণ সমাধান হল কোনও বিশ্বস্ত কেন্দ্রীয় কর্তৃপক্ষ, যেমন টাঁকশাল- যা দ্বিগুণ খরচের জন্য প্রতিটি লেনদেন পরীক্ষা করে। প্রতিটি লেনদেনের পরে, একটি নতুন কয়েন ইস্যু করার জন্য কয়েনটি অবশ্যই মিন্টে ফেরত দিতে হবে, এবং শুধুমাত্র মিন্ট থেকে সরাসরি জারি করা কয়েনগুলি দ্বিগুণ খরচ হবে না বলে বিশ্বাস করা হয়। এই সমাধানের সমস্যা হল যে, পুরো অর্থ ব্যবস্থার ভাগ্য নির্ভর করে সেই কোম্পানির উপরে, যে মিন্ট চালাচ্ছে। প্রতিটি লেনদেন তাদের মাধ্যমে যেতে হবে, ঠিক একটি ব্যাঙ্কের মতো।

প্রাপকদের ক্ষেত্রে, আমাদের কোনও একটি উপায়ে জানতে হবে যে পূর্ববর্তী মালিকরা আগের কোনও লেনদেনে স্বাক্ষর করেছেন কিনা। আমাদের উদ্দেশ্য হল, প্রথম দিকের লেনদেনটিই গণনা করা, তাই আমরা পরবর্তীতে দ্বিগুণ-ব্যয় করার প্রচেষ্টার বিষয়ে মাথাব্যথা করি না। কোনও একটি লেনদেন না হওয়ার বিষয়টি নিশ্চিত করার একমাত্র উপায় হল সমস্ত লেনদেন সম্পর্কে সচেতন হওয়া। মিন্ট ভিত্তিক মডেলে, মিন্ট সমস্ত লেনদেন সম্পর্কে অবগত ছিল এবং সিদ্ধান্ত গ্রহণ করত, যে কোনটি প্রথমে আসবে। একটি বিশ্বস্ত পক্ষ ছাড়া এটি সম্পন্ন করার জন্য লেনদেনগুলি অবশ্যই সর্বজনীনভাবে ঘোষণা করা উচিত, এবং আমাদের একটি এমন সিস্টেমের প্রয়োজন যেখানে অংশগ্রহণকারীরা যে ক্রমে টাকা পেয়েছিলেন- তার একক ইতিহাসে সম্মত থাকবেন। প্রাপকের কাছেও প্রমাণ থাকা প্রয়োজন যে প্রতিটি লেনদেনের সময়, বেশিরভাগ নোড তার প্রথম প্রাপ্তির ক্ষেত্রে সম্মত

হয়েছিল।

৩. টাইমস্ট্যাম্প সার্ভার

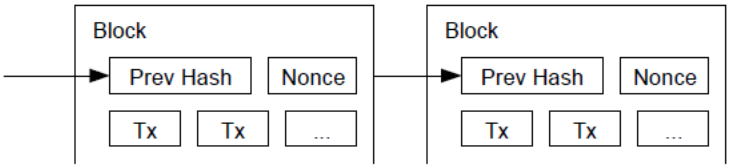
আমরা যে সমাধানের প্রস্তাব করেছি তা একটি টাইমস্ট্যাম্প সার্ভার দিয়ে শুরু হয়। একটি টাইমস্ট্যাম্প সার্ভার টাইমস্ট্যাম্প করার জন্য আইটেমগুলির এক-একটি ব্লকের একটি হ্যাশ নিয়ে কাজ করে এবং হ্যাশগুলিকে ব্যাপকভাবে প্রকাশ করে, যেমন একটি সংবাদপত্র বা ইউজ়নেট পোস্টে [২-৫] করা হয়। টাইমস্ট্যাম্প অবশ্যই হ্যাশে প্রবেশ করার ক্ষেত্রে প্রমাণ করে যে সেই সময়ে ওই ডেটা বিদ্যমান ছিল। স্পষ্টতই, প্রতিটি টাইমস্ট্যাম্প তার হ্যাশে পূর্ববর্তী টাইমস্ট্যাম্প অন্তর্ভুক্ত করে একটি শৃঙ্খল গঠন করে, প্রতিটি অতিরিক্ত টাইমস্ট্যাম্প তার আগেরগুলিকে আরও শক্তিশালী করে।



৪. প্রুফ-অফ-ওয়ার্ক

পিয়ার-টু-পিয়ার ভিত্তিতে বিতরণ করা একটি টাইমস্ট্যাম্প সার্ভার বাস্তবায়ন করতে, আমাদের সংবাদপত্র বা ইউজ়নেট পোস্টের পরিবর্তে অ্যাডাম ব্যাকের হ্যাশক্যাশ [৬]-এর মতো একটি প্রুফ-অফ-ওয়ার্ক সিস্টেম ব্যবহার করতে হবে। প্রুফ-অফ-ওয়ার্কে এমন একটি মানের জন্য স্ক্যান করা হয় যেটি SHA-256-এর মতো মান দিয়ে হ্যাশ করার সময় হ্যাশটি শূন্য বিটের একটি সংখ্যা দিয়ে শুরু হয়। প্রয়োজনীয় গড় কাজটি শূন্য বিটের প্রয়োজনে ব্যাখ্যায়িত এবং একটি একক হ্যাশ কার্যকর করার মাধ্যমে যাচাই করা যেতে পারে।

আমাদের টাইমস্ট্যাম্প নেটওয়ার্কের জন্য, ব্লকের হ্যাশকে প্রয়োজনীয় শূন্য বিট দেয় এমন একটি মান না পাওয়া পর্যন্ত আমরা ব্লকে একটি নন্স বৃদ্ধি করে কাজের প্রুফ-অফ-ওয়ার্ক বাস্তবায়ন করি। একবার প্রুফ-অফ-ওয়ার্ক নিয়ে সন্তুষ্ট হওয়ার লক্ষ্যে সিপিইউ-এর প্রচেষ্টা ব্যয় করা হয়ে গেলে কাজটি পুনরায় না করে ব্লকটি আর পরিবর্তন করা যাবে না। যেহেতু পরবর্তী ব্লকগুলি এর পরে শৃঙ্খলিত হয়, তাই একটি ব্লক পরিবর্তন করলে পরের সমস্ত ব্লক পুনরায় ঢেলে সাজাতে হয়।



সংখ্যাগরিষ্ঠের সিদ্ধান্ত গ্রহণে প্রতিনিধিত্ব নির্ধারণের সমস্যারও সমাধান করে প্রুফ-অফ-ওয়ার্ক । যদি সংখ্যাগরিষ্ঠ নোড এক-আইপি-অ্যাড্রেস-এক-ভোটের উপর ভিত্তি করে থাকে, তবে অনেকগুলি আইপি বরাদ্দ করতে সক্ষম যে কেউ তা বিকৃত হতে পারে। প্রুফ-অফ-ওয়ার্ক বলতে মূলত বোঝায় এক-সিপিইউ-এক-ভোট। সংখ্যাগরিষ্ঠ সিদ্ধান্তটির প্রতিনিধিত্ব করে দীর্ঘতম শৃঙ্খল এবং প্রুফ-অফ-ওয়ার্ক সর্বাধিক প্রচেষ্টা এখানেই বিনিয়োগ করা হয়। যদি বেশিরভাগ সিপিইউ শক্তি সৎ নোড দ্বারা নিয়ন্ত্রিত হয়, তাহলে সৎ শৃঙ্খল দ্রুততম বৃদ্ধি পাবে এবং যে কোনও প্রতিযোগী শৃঙ্খলকে ছাড়িয়ে যাবে। একটি অতীত ব্লক সংশোধন করতে, একজন আক্রমণকারীকে ব্লকের প্রুফ-অফ-ওয়ার্ক এবং তার পরে থাকা সমস্ত ব্লক পুনরায় করতে হবে এবং তারপরে সৎ

নোডগুলির কাজটি আটকাতে এবং অতিক্রম করতে হবে। আমরা পরে দেখাব যে, পরবর্তী ব্লকগুলি যোগ করার সঙ্গেই এই ধীরে আক্রমণকারীর সাফল্যের সম্ভাবনা দ্রুত হ্রাস পায়।
হার্ডওয়্যারের গতি বৃদ্ধি এবং সময়ের সঙ্গে সঙ্গে নোডগুলি চালনা করার আগ্রহের তারতম্যের জন্য হওয়া ক্ষতিপূরণ হিসেব কষতে প্রুফ-অফ-ওয়ার্কের সমস্যাকে একটি চলমান গড় দ্বারা নির্ধারিত হয়- যা প্রতি ঘন্টায় গড় সংখ্যক ব্লককে নিশানা করে। যদি তারা খুব দ্রুত বাড়তে থাকে, সমস্যা বৃদ্ধি পায়।

৫. নেটওয়ার্ক

নেটওয়ার্ক চালনা করার ধাপগুলি নিম্নরূপ:
১) নতুন লেনদেন সমস্ত নোডে সম্প্রচার করা হয়।
২) প্রতিটি নোড একটি ব্লকে নতুন লেনদেন সংগ্রহ করে।
৩) প্রতিটি নোড তার ব্লকের জন্য একটি অটাক্য প্রমাণ খুঁজে বের করার জন্য কাজ করে।
৪) যখন একটি নোড একটি প্রুফ-অফ-ওয়ার্ক খুঁজে পায়, তখন এটি সমস্ত নোডে ব্লকটি সম্প্রচার করে।
৫) নোডগুলি কেবল তখনই ব্লকটি গ্রহণ করে যদি এতে সমস্ত লেনদেন বৈধ হয় এবং ইতিমধ্যে ব্যয় না করা হয়।
৬) নোডগুলি পূর্ববর্তী হ্যাশ হিসাবে গৃহীত ব্লকের হ্যাশ ব্যবহার করে শৃঙ্খলে পরবর্তী ব্লক তৈরির মাধ্যমে ব্লকে তাদের গ্রহণযোগ্যতা প্রকাশ করে।

নোডগুলি সর্বদা দীর্ঘতম শৃঙ্খলটিকে সঠিক হিসাবে বিবেচনা করে এবং এটিকে প্রসারিত করার জন্য কাজ চালিয়ে যায়। যদি দুটি নোড একই সঙ্গে পরবর্তী ব্লকের বিভিন্ন সংস্করণ সম্প্রচার করে, কিছু নোড প্রথমে একটি বা অন্যটি গ্রহণ করতে পারে। সেক্ষেত্রে, তারা প্রাপ্ত প্রথম শৃঙ্খলাটি নিয়েই কাজ করে, তবে অন্য শাখাটি যদি দীর্ঘ হয়ে যায়, তাকেও সংরক্ষণ করে। পরবর্তী প্রুফ-অফ-ওয়ার্ক পাওয়া গেলে এবং একটি শাখা দীর্ঘ হয়ে গেলে টাই ভেঙে যাবে; যে নোডগুলি অন্য শাখায় কাজ করছিল সেগুলি তারপরে দীর্ঘতর শাখায় যুক্ত হয়ে যাবে।
নতুন লেনদেন সম্প্রচারগুলির সব নোডে পৌঁছানোর প্রয়োজন হয় না। যতক্ষণে তারা অনেক নোডে পৌঁছাবে, ততক্ষণে তারা একটি ব্লকে প্রবেশ করে যাবে। ব্লক সম্প্রচারগুলি ড্রপ হওয়া বার্তার প্রতিও সহনশীল। যদি একটি নোড কোনও ব্লক গ্রহণ না করে, সে ক্ষেত্রে পরবর্তী ব্লকটি গ্রহণ করার পরে অনুরোধ আসে এবং সে বুঝতে পারে যে একটি ব্লক বাদ পড়ে গিয়েছে।

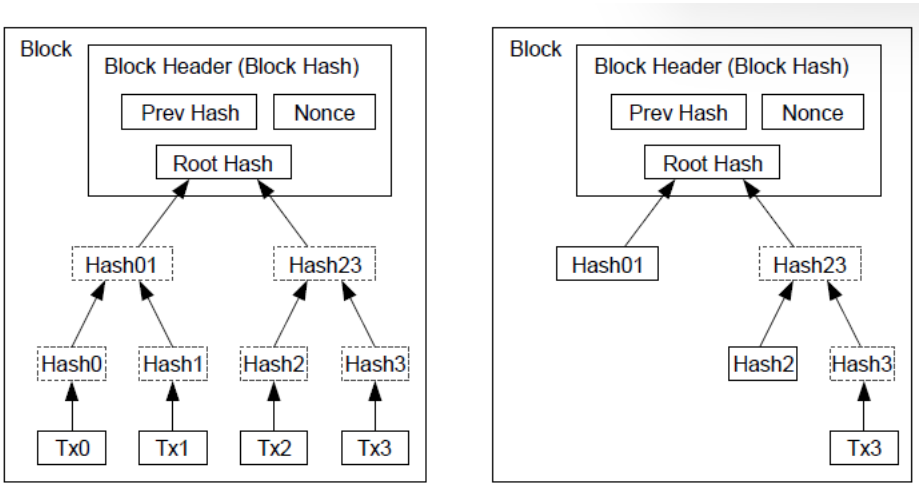
৬. ইনসেনটিভ

নিয়ম অনুসারে, একটি ব্লকে প্রথম লেনদেন হল সেই বিশেষ লেনদেন যা ব্লকের নির্মাতার মালিকানাধীন একটি নতুন কয়েন শুরু করে। এটি নেটওয়ার্ককে সমর্থন করতে নোডগুলির জন্য একটি প্রণোদনা বা ইনসেনটিভ যোগ করে এবং যেহেতু সেগুলি ইস্যু করার জন্য কোনও কেন্দ্রীয় কর্তৃপক্ষ নেই, তাই প্রাথমিকভাবে কয়েনগুলিকে প্রচলন করার জন্য বিতরণের একটি উপায় প্রদান করে। ক্রমাগত নতুন কয়েনের একটি স্থির সংযোজন সোনার খনির শ্রমিকদের সঞ্চালনে সোনা যোগ করার ক্ষেত্রে সম্পদ ব্যয় করার অনুরূপ। আমাদের ক্ষেত্রে, এটি সিপিইউ সময় এবং বিদ্যুতের ব্যয়।
ইনসেনটিভটি লেনদেন ফি দিয়েও অর্থায়ন করা যেতে পারে। যদি একটি লেনদেনের আউটপুট মান তার ইনপুট মানের থেকে কম হয়, তবে ওই পার্থক্য হল সেই লেনদেন ফি- যা লেনদেন পরিচালনকারী ব্লকের ইনেসনটিভ মূল্যে যোগ করা হয়। একবার পূর্বনির্ধারিত সংখ্যক কয়েন প্রচলনে প্রবেশ করলে, ইনসেনটিভ সম্পূর্ণরূপে লেনদেন ফিতে রূপান্তরিত হতে পারে এবং সম্পূর্ণ মুদ্রাস্ফীতি মুক্ত হতে পারে।

ইনসেনটিভ নোডকে সৎ থাকতে উৎসাহিত করতে সাহায্য করতে পারে। যদি একজন লোভী আক্রমণকারী সমস্ত সৎ নোডের চেয়ে বেশি সিপিইউ শক্তি একত্র করতে সক্ষম হয়, তাহলে সে তার সব প্রদত্ত অর্থ চুরি করে লোকেদের প্রতারণা করার জন্য এটি ব্যবহার করতে পারে বা নতুন কয়েন তৈরি করতে ব্যবহার করতে পারে। তার উচিত সিস্টেম এবং তার নিজের সম্পদের বৈধতাকে ক্ষুণ্ন করার চেয়ে নিয়ম মেনে খেলা যা আরও বেশি লাভজনক হতে পারে। এমন নিয়ম যা তাকে অন্য সবার চেয়ে বেশি নতুন কয়েন দিয়ে সমর্থন করে।

৭ ডিস্ক স্পেস পুনরুদ্ধার করা

একটি কয়েনের সর্বশেষ লেনদেনটি পর্যাপ্ত ব্লকের নিচে চাপা পড়লে, ডিস্কের স্থান বাঁচাতে খরচ করা লেনদেন বাতিল করা যেতে পারে। ব্লকের হ্যাশ না ভেঙে এটি সহজতর করার জন্য, লেনদেনগুলিকে একটি মার্কলে ট্রিতে হ্যাশ করা হয় [৭][২][৫], ব্লকের হ্যাশের মধ্যে শুধুমাত্র রুটটি অন্তর্ভুক্ত থাকে। পুরানো ব্লকগুলি অপ্রয়োজনীয় শাখাগুলো বন্ধ করে দিয়ে কম্প্যাক্ট করা যেতে পারে। অভ্যন্তরীণ হ্যাশগুলি সংরক্ষণ করার প্রয়োজন নেই।

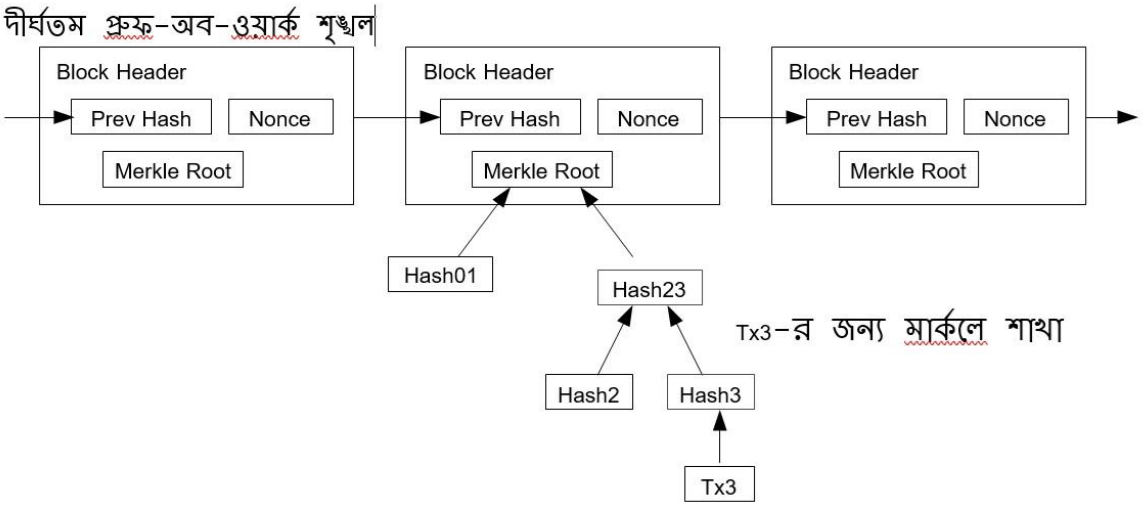


Transactions Hashed in a Merkle Tree | After Pruning Tx0-2 from the Block

কোনও লেনদেন ছাড়া একটি ব্লক হেডার প্রায় ৮০ বাইট আকারের হয়। যদি আমরা মনে করি প্রতি ১০ মিনিটে ব্লক তৈরি হয়, ৮০ বাইট * ৬ * ২৪ * ৩৬৫ = ৪.২ এমবি প্রতি বছর। ২০০৮ সালের হিসাবে কম্পিউটার সিস্টেমগুলি সাধারণত ২জিবি র‍্যামের সঙ্গে বিক্রি হয় এবং মুরের আইন প্রতি বছর ১.২ জিবির বর্তমান বৃদ্ধির পূর্বাভাস দেয়, তাই ব্লক হেডারগুলি মেমরিতে রাখতে গেলেও স্টোরেজের কোনও সমস্যা হবে না।

৮. সরলীকৃত অর্থপ্রদান যাচাইকরণ

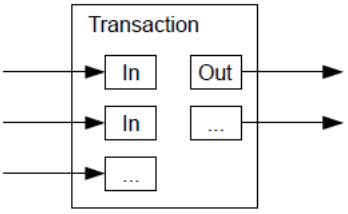
একটি সম্পূর্ণ নেটওয়ার্ক নোড না চালিয়েও অর্থপ্রদান যাচাই করা সম্ভব। একজন ব্যবহারকারী যতক্ষণ না নিশ্চিত হন যে তার কাছে দীর্ঘতম শৃঙ্খল রয়েছে এবং লেনদেনটিকে টাইমস্ট্যাম্প করা ব্লকের সঙ্গে যুক্ত করে মার্কলে শাখাটি পাচ্ছেন, ততক্ষণ তাঁকে শুধুমাত্র দীর্ঘতম প্রুফ-অফ-ওয়ার্ক শৃঙ্খলের ব্লক হেডারগুলির একটি অনুলিপি রাখতে হবে, যা তিনি নেটওয়ার্ক নোডগুলি অনুসন্ধান করে পেতে পারেন। তিনি নিজের জন্য লেনদেনটি পরীক্ষা করতে পারবেন না, তবে এটিকে শৃঙ্খলের একটি জায়গায় যুক্ত করার মাধ্যমে, তিনি দেখতে পারেন যে নেটওয়ার্ক নোড সেটিকে গ্রহণ করেছে, এবং পরবর্তীকালে যুক্ত হওয়া ব্লকগুলিও নিশ্চিত করে যে নেটওয়ার্ক এটি গ্রহণ করে নিয়েছে।



সৎ নোডগুলি যতক্ষণ পর্যন্ত নেটওয়ার্ক নিয়ন্ত্রণ করে ততক্ষণ যাচাইকরণ নির্ভরযোগ্য, তবে নেটওয়ার্কটি আক্রমণকারীর দ্বারা দখল হয়ে গেলে এটি খুবই ঝুঁকিপূর্ণ হয়ে যায়। যদিও নেটওয়ার্ক নোডগুলি নিজেদের জন্য লেনদেনগুলি যাচাই করতে পারে, তবে সেই সরল পদ্ধতিকে আক্রমণকারী নেটওয়ার্কের তৈরি ছদ্মবেশী লেনদেন দ্বারা ঠকানো সম্ভব। এর বিরুদ্ধে রক্ষা করার একটি কৌশল হ'ল যখনই নেটওয়ার্ক নোডগুলি একটি অবৈধ ব্লক শনাক্ত করে, ব্যবহারকারীর সফ্টওয়্যারকে সম্পূর্ণ ব্লক ডাউনলোড করার জন্য অনুরোধ করে এবং অসঙ্গতি নিশ্চিত করার জন্য লেনদেনগুলিকে সতর্ক করে, তখনই সতর্ক হয়ে যাওয়া যে ব্যবসাগুলি ঘন ঘন পেমেন্ট পায় তারা সম্ভবত আরও স্বাধীন নিরাপত্তা এবং দ্রুত যাচাইকরণের জন্য তাদের নিজস্ব নোড চালাতে আগ্রহী হবে।

৯. সমন্বয় এবং বিভাজন মান

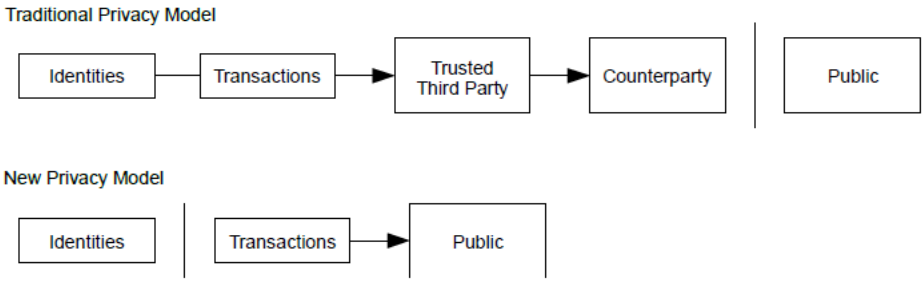
যদিও স্বতন্ত্রভাবে কয়েন বা কয়েন পরিচালনা করা সম্ভব, তবে একটি স্থানান্তরের প্রতিটি সেন্টের জন্য একটি পৃথক লেনদেন করা কঠিন হবে। মানকে বিভক্ত এবং একত্রিত করার অনুমতি দেওয়ার জন্য, লেনদেনে একাধিক ইনপুট এবং আউটপুট থাকে। সাধারণত বৃহত্তর পূর্ববর্তী লেনদেন থেকে হয় একটি একক ইনপুট বা একাধিক ইনপুট ছোট পরিমাণে একত্রিত হয়, এবং সর্বাধিক দুটি আউটপুট: একটি অর্থপ্রদানের জন্য, এবং অন্যটি, যদি ফেরতযোগ্য অর্থ থাকে তা প্রেরকের কাছে ফেরত পাঠাতে ব্যবহৃত হয়।



এটি মনে রাখতে হবে যে, ফ্যান-আউট, অর্থাৎ যেখানে একটি লেনদেন বেশ কয়েকটি লেনদেনের উপর নির্ভর করে এবং সেই লেনদেনগুলি আরও অনেকের উপর নির্ভর করে, সেখানে কোনও সমস্যা নেই। একটি লেনদেনের ইতিহাসের একটি সম্পূর্ণ স্বতন্ত্র কপি বের করারও প্রয়োজন নেই।

১০. গোপনীয়তা

প্রথাগত ব্যাঙ্কিং মডেলগুলি জড়িত পক্ষ এবং বিশ্বস্ত তৃতীয় পক্ষের তথ্যের অ্যাক্সেস সীমিত করে গোপনীয়তার একটি স্তর অর্জন করে। সমস্ত লেনদেন ঘোষণা করার প্রয়োজনীয়তা সর্বজনীনভাবে এই পদ্ধতিটিকে নিষ্ক্রিয় করে দেয়, তবে এখনও অন্য জায়গায় তথ্যের প্রবাহকে ভেঙে দিয়ে গোপনীয়তা বজায় রাখা যেতে পারে, যথা: সর্বজনীন কী গুলি বেনামী রেখে। জনসাধারণ দেখতে পারে যে কেউ অন্য কাউকে নির্দিষ্ট পরিমাণে অর্থ পাঠাচ্ছে, কিন্তু ওই লেনদেনের কোনও তথ্য কারও সঙ্গে যুক্ত হয় না। এটি স্টক এক্সচেঞ্জ দ্বারা প্রকাশিত তথ্যের স্তরের অনুরূপ, যেখানে একটি "টেপ" অর্থাৎ পৃথক ট্রেডের সময় এবং আকার সর্বজনীন করা হয়, কিন্তু পক্ষগুলি কারা ছিল- তা প্রকাশ করা হয় না।



প্রথাগত গোপনীয়তার মডেল

নতুন গোপনীয়তা মডেল

একটি অতিরিক্ত ফায়ারওয়াল হিসাবে, প্রতিটি লেনদেনের জন্য একটি নতুন কী-এর জোড়া ব্যবহার করা উচিত যাতে সেগুলিকে একটি সাধারণ মালিকের সঙ্গে সংযুক্ত না করতে হয়। মাল্টি-ইনপুট লেনদেনের সঙ্গে কিছু লিঙ্কিং এখনও অনিবার্য, যেখানে বাধ্য হয়ে এই তথ্য প্রকাশ করা হয় যে তাদের ইনপুটগুলি একই মালিকের মালিকানাধীন। ঝুঁকি হল, যে যদি একটি চাবির মালিক প্রকাশিত হয়, লিঙ্ক করা একই মালিকের অন্তর্গত অন্যান্য লেনদেনও প্রকাশ হয়ে যেতে পারে।

১১. গণনা

আমরা যদি ধরে নিই, কোনও এক আক্রমণকারী সৎ শৃঙ্খলের চেয়ে দ্রুত একটি বিকল্প শৃঙ্খল তৈরি করার চেষ্টা করছে এমনকী যদি তা তৈরিও করে ফেলে, তাহলেও এটি সিস্টেমটিকে নির্বিচারে পরিবর্তনের জন্য উন্মুক্ত করে দেয় না। অর্থাৎ কাল্পনিক মূল্য তৈরি করা বা আদতে আক্রমণকারীর নয় এমন অর্থ নিয়ে নেওয়া এত সহজ নয়। নোডগুলি অর্থপ্রদানের ক্ষেত্রে একটি অবৈধ লেনদেন গ্রহণ করবে না এবং সৎ নোডগুলি কখনই অবৈধ লেনদেন ধারণকারী ব্লককে স্বীকার করবে না। অবশ্য একজন আক্রমণকারী তার সম্প্রতি ব্যয় করা অর্থ ফেরত নিতে তার নিজের লেনদেনে একটি পরিবর্তন করার চেষ্টা করতে পারে।
সৎ শৃঙ্খল এবং আক্রমণকারী শৃঙ্খলের মধ্যে প্রতিযোগিতাকে একটি 'বাইনমিয়াল র‍্যান্ডম ওয়াক' হিসাবে চিহ্নিত করা যেতে পারে। সাফল্যের ক্ষেত্রে সৎ শৃঙ্খলটি একটি ব্লক প্রসারিত হয়, এটির লিড +১ বৃদ্ধি পায় এবং ব্যর্থতা হল যেখানে আক্রমণকারীর শৃঙ্খলটি একটি ব্লক প্রসারিত হয় ও ব্যবধান -১ কমিয়ে আনে।

পিছনে পড়ে থাকা আক্রমণকারীর দৌড়ে এসে ফারাক ঘুচিয়ে দেওয়ার সম্ভাবনা 'গ্যাম্বলার্স রুইন প্রবলেম' বা জুয়াড়ির ধ্বংসাত্মক সমস্যার সঙ্গে সাদৃশ্যপূর্ণ। মনে করুন, সীমাহীন ক্রেডিট-সহ একজন জুয়াড়ি একটি ঘাটতি থেকে কাজ শুরু করল এবং ব্যবধান ঘোচানোর লক্ষ্যে পৌঁছানোর চেষ্টা করতে সম্ভাব্য সীমাহীন সুযোগ নেওয়ার চেষ্টা চালিয়ে গেল। আক্রমণকারী কখন সমান-সমান স্থানে বা ব্রেকইভেনে পৌঁছাতে পারে, বা কখন সৎ শৃঙ্খলটি পার করে ফেলতে পারে আমরা তার সম্ভাব্যতা গণনা করতে পারি নিম্নরূপে [৮]:

p = probability an honest node finds the next block
q = probability the attacker finds the next block
q_z = probability the attacker will ever catch up from z blocks behind

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

আমরা p > q অনুমান করে নিলে, আক্রান্ত হওয়ার সম্ভাব্যতা দ্রুত হ্রাস পায় কারণ আক্রমণকারীকে তখন যতগুলি ব্লক অতিক্রম করতে হবে- তার সংখ্যা বৃদ্ধি পায়। এই প্রতিকূল পরিস্থিতিতে যদি সে আগে থেকে সুযোগমতো বড় ঝাঁপ না দিতে পারে, তবে সে আরও পিছিয়ে পড়বে এবং সেই সঙ্গে তার সম্ভাবনা অদৃশ্য হয়ে যাবে।

আমরা এখন বিবেচনা করে দেখি, প্রেরক আর লেনদেন পরিবর্তন করতে পারবেন না- তা নিশ্চিত হওয়ার জন্য একটি নতুন লেনদেনের প্রাপককে কতক্ষণ অপেক্ষা করতে হতে পারে। আমরা অনুমান করি যে, প্রেরক একজন আক্রমণকারী যে প্রাপককে অর্থ প্রদান করার বিষয়টি খানিকটা সময়ের জন্য বিশ্বাস করাতে চাইছে এবং কিছু সময় অতিবাহিত হওয়ার পরে ওই অর্থ সে নিজের কাছে ফেরত নিয়ে আসবে। যখন এটি ঘটবে তখন প্রাপককে সতর্ক করা হবে, কিন্তু আক্রমণকারী প্রেরক আশা করবে যে প্রাপক যতক্ষণে সতর্ক হবেন, ততক্ষণে অনেক দেরি হয়ে গিয়েছে।

প্রাপক একজোড়া নতুন কী তৈরি করেন এবং স্বাক্ষর করার কিছুক্ষণ আগে প্রেরককে সর্বজনীন কী প্রদান করেন। এটি ক্রমাগতভাবে কাজ করে প্রেরককে সময়ের আগে ব্লকের একটি শৃঙ্খল প্রস্তুত করতে বাধা দেয়, যদি না সে যথেষ্ট ভাগ্যবান হয় ও তার চেয়ে বেশি দ্রুত এগিয়ে গিয়ে সেই মুহূর্তেই লেনদেন সম্পাদন করতে সক্ষম হয়। একবার লেনদেন পাঠানো হলে, অসাধু প্রেরক তার লেনদেনের একটি বিকল্প সংস্করণ ধারণকারী সমান্তরাল শৃঙ্খলে গোপনে কাজ শুরু করে।

যতক্ষণ না লেনদেনটি একটি ব্লকে যোগ করা হয় এবং z ব্লকগুলি এর পরে সংযুক্ত করা হয়, ততক্ষণ প্রাপক অপেক্ষা করেন। আক্রমণকারীর ঠিক কতটা অগ্রগতি হয়েছে তার সঠিক মাত্রা তিনি জানেন না, তবে অনুমান করেন যে সৎ ব্লকগুলি প্রতি ব্লকের ক্ষেত্রে গড় প্রত্যাশিত সময় নিয়েছে। আক্রমণকারীর সম্ভাব্য অগ্রগতি একটি 'পয়সন বিতরণ' মেনে হবে যার প্রত্যাশিত মূল্য হবে:

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

To get the probability the attacker could still catch up now, we multiply the Poisson density for each amount of progress he could have made by the probability he could catch up from that point:

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

Rearranging to avoid summing the infinite tail of the distribution...

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

Converting to C code...

```
#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;
}
```

Running some results, we can see the probability drop off exponentially with 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
```

Solving for P less than 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
```

১২. উপসংহার

আমরা বিশ্বাসের উপর নির্ভর না করে বৈদ্যুতিক লেনদেনের জন্য একটি সিস্টেমের প্রস্তাব পেশ করেছি। যেখানে ডিজিটাল স্বাক্ষর থেকে তৈরি কয়েনের স্বাভাবিক কাঠামো দিয়ে আমরা কাজ শুরু করি, যা মালিকানার ক্ষেত্রে দৃঢ় নিয়ন্ত্রণ প্রদান করে। কিন্তু দ্বিগুণ খরচ রোধ করার উপায় ছাড়া এটি অসম্পূর্ণ। সেই সমস্যার সমাধান হিসেবে, আমরা লেনদেনের একটি সর্বজনীন ইতিহাস রেকর্ড করতে প্রুফ-অফ-ওয়ার্ক ব্যবহার করে একটি পিয়ার-টু-পিয়ার নেটওয়ার্কের প্রস্তাব দিয়েছি। এ ক্ষেত্রে সৎ নোডগুলি বেশিরভাগ সিপিইউ শক্তি নিয়ন্ত্রণ করলে আক্রমণকারীর পক্ষে তা পরিবর্তন করা দ্রুত গণনাগতভাবে অবাস্তব হয়ে দাঁড়ায়। নেটওয়ার্কটি তার গঠনহীন সরলতার জোরেই শক্তিশালী। নোডগুলি সামান্য সমন্বয়ের সঙ্গে একত্রে কাজ করে। যেহেতু বার্তাগুলি কোনও নির্দিষ্ট জায়গায় পাঠানো হয় না এবং শুধুমাত্র সর্বোত্তম প্রচেষ্টার ভিত্তিতে বিতরণ করা প্রয়োজন তাই তাদের চিহ্নিত করারও দরকার পড়ে না। নোডগুলি ইচ্ছামত নেটওয়ার্ক ছেড়ে যেতে এবং পুনরায় যোগদান করতে পারে। চলে যাওয়ার সময় যা ঘটেছিল তার প্রমাণ হিসাবে তারা প্রুফ-অব-ওয়ার্ক শৃঙ্খলকে গ্রহণ করে। তারা তাদের সিপিইউ শক্তির সাহায্যে ভোট

দেয়, বৈধ ব্লকগুলি প্রসারিত করার মাধ্যমে তাদের গ্রহণযোগ্যতাকে প্রকাশ করে এবং অবৈধ ব্লকগুলিকে প্রত্যাখ্যান করে তাদের নিয়ে কাজ করতে অস্বীকার করে। যে কোনও প্রয়োজনীয় নিয়ম এবং ইনসেনটিভ এই সর্বসম্মত প্রক্রিয়ার সঙ্গে প্রয়োগ করা যেতে পারে।

References

[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.