பிட்காயின்: சக நபர்களுக்கிடையே மின்னணுவியல் முறையில் கேஷ் (நாணயப் பணம்) சிஸ்டம்

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

சுருக்கம்: மின்னணுவியல் முறையில் நாணய பணத்தின் சக நபர்களுக்கிடையே (peer-to-peer) செயல் முறையானது, நிதி நிறுவனம் வழியான பரிவர்த்தனையின்றி ஒரு நபரிடமிருந்து மற்றொரு நபருக்கு நேரடியாக ஆன்லைன் பேமென்ட்களை (பணம் செலுத்தல்களை) அனுப்ப அனுமதிக்கும். டிஜிட்டல் கையொப்பங்கள் இத்தீர்வின் ஒரு பகுதியை வழங்குகின்றன; ஆனால் இரட்டைச் செலவினத்தைத் தடுக்க நம்பகமான ஒரு மூன்றாம் தரப்பு அப்போதும் தேவைப்படுமானால், முக்கிய நன்மைகள் இழக்கப்படும். சக நபர்களுக்கிடையிலான நெட்வொர்க்கைப் பயன்படுத்தி இரட்டைச் செலவுச் சிக்கலுக்கு ஒரு தீர்வை நாங்கள் முன்மொழிகிறோம். நேர முத்திரை (டைம்ஸ்டாம்ப்) பரிவர்த்தனைகளை ஹாஷ்-அடிப்படையிலான ப்ரூஃப்-ஆஃப்-வொர்க் தொடரில் ஹாஷ் செய்வதன் மூலம், வேலைச் சான்றுகளை மீண்டும் செய்யாமல் மாற்ற முடியாத ஒரு பதிவை நெர்வொர்க்கை உருவாக்குகிறது.

மிக நீளமான சங்கிலி, நிகழ்வுகளின் வரிசையின் சான்றாக இருப்பது மட்டுமல்லாமல், CPU சக்தியின் மிகப்பெரிய தொகுப்பிலிருந்து இது வந்தது என்பதற்கான சான்றாகவும் உள்ளது. நெட்வொர்க்கைத் தாக்க ஒத்துழைக்காத கணுக்களால் பெரும்பாலான CPU சக்தி கட்டுப்படுத்தப்படும் வரை, அவை மிக நீளமான சங்கிலியை உருவாக்கும் மற்றும் அவுட்பேஸ் தாக்குபவர்களை முறியடிக்கும். நெட்வொர்க்கிற்கே குறைந்தபட்ச கட்டமைப்பு தேவைப்படுகிறது. மெசேஜ்கள் (செய்திகள்) சிறந்த முயற்சியின் அடிப்படையில் ஒலிபரப்பப்படுகின்றன; மேலும் கணுக்கள் (Nodes) தாங்கள் விரும்பியபடி பிணையத்தை விட்டு வெளியேறலாம் மற்றும் அவர்கள் வெளியேறி சென்றிருந்தபோது, என்ன நடந்தது என்பதற்கான சான்றாக மிக நீளமான பணிச் சங்கிலி சான்றை ஏற்றுக்கொண்டு மீண்டும் இணையலாம்.

1. அறிமுகம்

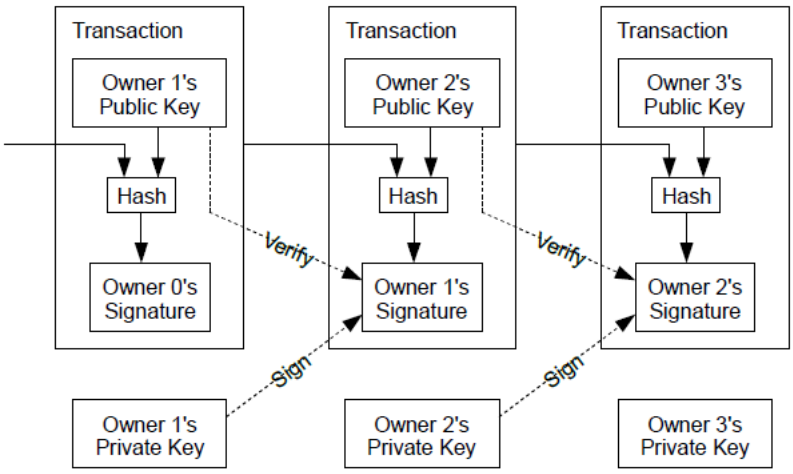
இணையத்தில் மேற்கொள்ளப்படும் வணிகமானது, மின்னணு பேமெண்ட்களை (கொடுப்பனவுகள்) செயல்படுத்த நம்பகமான மூன்றாம் தரப்பினராகச் செயல்படும் நிதி நிறுவனங்களை மட்டுமே நம்பியிருக்கிறது. பெரும்பாலான பரிவர்த்தனைகளுக்கு இந்த அமைப்பு முறை நன்றாக செயல்படுகிறது என்றாலும், நம்பிக்கை அடிப்படையிலான மாதிரியின் உள்ளார்ந்த பலவீனங்களால் அது இன்னும் பாதிப்பிற்கு ஆளாகிறது. தகராறுகளை மத்தியஸ்தம் செய்வதை நிதி நிறுவனங்களால் தவிர்க்க முடியாது என்பதால், திரும்பவும் முழுமையாக மாற்ற இயலாத பரிவர்த்தனைகள் என்பவை உண்மையில் சாத்தியமில்லை. மத்தியஸ்தத்தின் செலவு பரிவர்த்தனை செலவுகளை அதிகரிக்கிறது, குறைந்தபட்ச யதார்த்தமான பரிவர்த்தனை அளவைக் கட்டுப்படுத்துகிறது மற்றும் சிறிய, சாதாரண பரிவர்த்தனைகளுக்கான சாத்தியத்தை நீக்கிவிடுகிறது. மேலும் திரும்பவும் மாற்ற இயலாத சேவைகளுக்கு, திரும்பவும் மாற்ற இயலாத பேமெண்ட்களை செய்யும் திறனை இழப்பதில் பரந்த செலவு உள்ளது. திரும்பவும் மாற்றிக் கொள்வதற்கான சாத்தியக்கூறின் மூலம் நம்பிக்கைக்கான அவசியம் பரவுகிறது. வணிகர்கள் தங்கள் வாடிக்கையாளர்களைப் பற்றி அதிக எச்சரிக்கையோடு இருப்பார்கள்; அவர்களுக்குத் தேவைப்படுவதை விட கூடுதல் தகவலுக்காக வாடிக்கையாளர்களை தொந்தரவு செய்யக்கூடும். ஒரு குறிப்பிட்ட சதவீத அளவிற்கு மோசடி தவிர்க்க முடியாதது ஏற்றுக்கொள்ளப்படுகிறது. இந்த செலவுகளையும், மற்றும் பணம் செலுத்தலில் நிச்சயமற்ற தன்மைகளையும் பணத்தாள்களை (கரன்சி) நேரடியாகப் பயன்படுத்துவதன் மூலம் தவிர்க்க முடியும். ஆனால் நம்பகமான ஒரு தரப்பு இல்லாமல் தகவல்தொடர்பு சேனல் மூலம் பணம் செலுத்துவதற்கான இயங்குமுறைகள் எதுவும் தற்போது இல்லை.

நம்பிக்கைக்கு பதிலாக கிரிப்டோகிராஃபிக் ஆதாரத்தை அடிப்படையாகக் கொண்ட மின்னணு பணம் செலுத்தல் முறையே இப்போது தேவைப்படுகிறது. இது, நம்பகமான மூன்றாம்

தரப்பு நபருக்கான தேவையின்றி விருப்பமுள்ள எந்த இரு தரப்பினரையும் ஒருவருக்கொருவர் நேரடியாக பரிவர்த்தனை செய்ய அனுமதிக்கிறது. திரும்பி மாற்றிக்கொள்வதற்கு கணக்கீட்டு ரீதியாக நடைமுறைக்கு சாத்தியமற்ற பரிவர்த்தனைகள் விற்பனையாளர்களை மோசடியிலிருந்து பாதுகாக்கும்; வாங்குபவர்களைப் பாதுகாக்க வழக்கமான எஸ்க்ரோ வழிமுறைகளை எளிதாக செயல்படுத்தலாம். இந்த ஆவணத்தில் பரிவர்த்தனைகளின் காலவரிசை முறையில் கணக்கீட்டு ஆதாரச் சான்றை உருவாக்க, சக நபர்களுக்கிடையே விநியோகிக்கப்பட்ட நேர முத்திரை சர்வரைப் பயன்படுத்தி இரட்டைச் செலவு பிரச்சனைக்கு ஒரு தீர்வை நாங்கள் முன்மொழிகிறோம். நேர்மையான கணுக்கள் கூட்டாக அதிக CPU சக்தியைக் கட்டுப்படுத்தும் வரை, தாக்குபவர் கணுக்களின் கூட்டுக்குழுவைக் காட்டிலும் இந்த சிஸ்டம் அதிக பாதுகாப்பாக இருக்கும்.

2. பரிவர்த்தனைகள்

ஒரு மின்னணு நாணயத்தை (எலக்ட்ரானிக் காயின்) டிஜிட்டல் கையொப்பங்களின் சங்கிலியாக நாம் வரையறை செய்கிறோம். ஒவ்வொரு உரிமையாளரும் முந்தைய பரிவர்த்தனையின் ஹாஷ் மற்றும் அடுத்த உரிமையாளரின் பொது கீயை டிஜிட்டல் முறையில் கையொப்பமிட்டு நாணயத்தின் முடிவில் அவற்றைச் சேர்ப்பதன் மூலம் நாணயத்தை அடுத்தவருக்கு மாற்றுகிறார். உரிமைத்துவ சங்கிலியைச் சரிபார்க்க பணம் பெறுபவர் கையொப்பங்களைச் சரிபார்த்து உறுதிசெய்ய முடியும்.



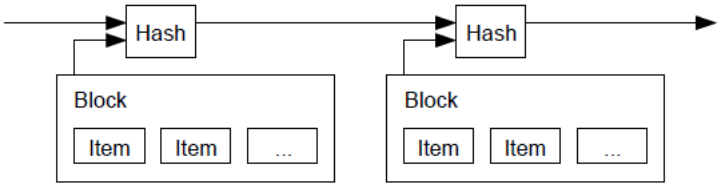
ஆனால், பிரச்சனை என்னவென்றால், உரிமையாளர்களில் ஒருவர் நாணயத்தை இருமுறை செலவழிக்கவில்லை என்பதை பணம் பெறுபவர் சரிபார்க்க முடியாது. ஒவ்வொரு பரிவர்த்தனையையும் இரட்டைச் செலவுக்காகச் சரிபார்க்கும் நம்பகமான மத்திய அதிகார அமைப்பு அல்லது நாணய தயாரிப்பாளர் / மின்ட் - ஐ அறிமுகப்படுத்துவதே பொதுவான தீர்வாக இருக்கும். ஒவ்வொரு பரிவர்த்தனைக்குப் பிறகும், புதிய நாணயத்தை வெளியிடுவதற்காக மின்ட்-க்கு அந்நாணயம் திருப்பி அனுப்பப்பட வேண்டும், மேலும் மின்ட் - லிருந்து நேரடியாக வெளியிடப்பட்ட நாணயங்கள் மட்டுமே இரட்டிப்புச் செலவு செய்யப்படாது என நம்பப்படுகிறது. இந்தத் தீர்வில் உள்ள சிக்கல் என்னவென்றால், ஒட்டுமொத்த பண அமைப்பும் மின்ட் – ஐ நடத்தும் நிறுவனத்தை சார்ந்திருக்கும். ஒரு வங்கியில் நிகழ்வதைப் போலவே ஒவ்வொரு பரிவர்த்தனையும் அந்நிறுவனத்தின் வழியாகவே சென்றாக வேண்டும்.

மேனாள் உரிமையாளர்கள் முந்தைய பரிவர்த்தனைகளில் கையொப்பமிடவில்லை என்பதை பணம் பெறுபவர் தெரிந்துகொள்ள நமக்கு ஒரு வழிமுறை தேவைப்படுகிறது. நமது நோக்கங்களுக்காக, மிக முந்தைய பரிவர்த்தனையே முக்கியமானது. எனவே, இரட்டை செலவிடலுக்கான பிந்தைய முயற்சிகளைப் பற்றி நாம் கவலைப்பட மாட்டோம். ஒரு பரிவர்த்தனை இல்லாததை உறுதிப்படுத்த ஒரே வழி, அனைத்து பரிவர்த்தனைகளையும் பற்றி அறிந்திருப்பதுதான். மின்ட் அடிப்படையிலான மாதிரியில், அனைத்து பரிவர்த்தனைகளையும் மின்ட் அறிந்திருந்தது மற்றும் எது முதலில் வந்தது என்பதை அது முடிவு செய்தது. நம்பகமான ஒரு தரப்பு இல்லாமல் இதைச் செய்வதற்கு பரிவர்த்தனைகள் பகிரங்கமாக அறிவிக்கப்பட வேண்டும் [1]; மேலும் பங்கேற்பாளர்கள் பெறப்பட்ட வரிசையின் ஒற்றை வரலாற்றை ஒப்புக்கொள்வதற்கு

நமக்கு ஒரு அமைப்பு தேவைப்படுகிறது. ஒவ்வொரு பரிவர்த்தனை நடைபெறும் நேரத்திலும், இதுதான் முதலில் பெறப்பட்டது என்று பெரும்பான்மையான கணுக்கள் ஒப்புக்கொண்டதற்கான ஆதாரம் பணம் பெறுபவருக்கு தேவைப்படுகிறது.

3. நேர முத்திரை சர்வர் (சேவையகம்)

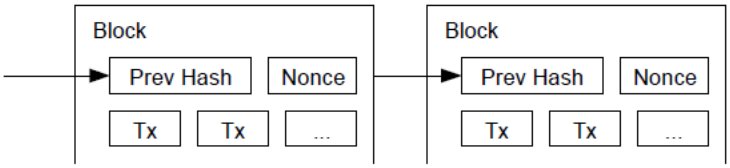
நாங்கள் பரிந்துரைக்கும் தீர்வு, நேர முத்திரை சர்வரிலிருந்து தொடங்குகிறது. டைம்ஸ்டாம்ப் சர்வர் நேர முத்திரையிடப்பட வேண்டிய உருப்படிகளின் ஒரு ஹாஷை எடுத்து, செய்தித்தாள் அல்லது யூஸ்நெட் போஸ்ட் [2-5] போன்றவற்றில் ஹாஷை பரவலாக பிரசுரிப்பதன் மூலம் செயல்படுகிறது. ஹாஷில் நுழைவதற்கு, அந்த நேரத்தில் தரவு இருந்திருக்க வேண்டும் என்பதை நேரமுத்திரை நிரூபிக்கிறது. ஒவ்வொரு நேர முத்திரையும் அதன் ஹாஷில் முந்தைய நேர முத்திரையை உள்ளடக்கி, ஒரு சங்கிலியை உருவாக்குகிறது, ஒவ்வொரு கூடுதல் நேரமுத்திரையும் அதற்கு முன் உள்ள நேர முத்திரைகளை மேலும் வலுப்படுத்தும்.



4. பணிக்கான சான்று

விநியோகிக்கப்பட்ட நேர முத்திரை சர்வரை சக நபர்களுக்கிடையிலான அடிப்படையில் செயல்படுத்த, செய்தித்தாள் அல்லது யூஸ்நெட் போஸ்ட்களுக்குப் பதிலாக ஆடம் பேக் - ன் ஹாஷ்காஷ் [6] போன்ற பணிச் சான்று முறையை நாம் பயன்படுத்துவது அவசியம். ஹாஷ் செய்யப்பட்ட மதிப்பை உள்வாங்குதலை (ஸ்கேன் செய்வதை) பணிக்கான சான்று உள்ளடக்குகிறது, பல பூஜ்ஜிய பிட்களுடன் ஹாஷ் தொடங்குகிற SHA-256 போன்றது இது. தேவைப்படும் சராசரி பணியானது, தேவைப்படும் பூஜ்ஜிய பிட்களின் எண்ணிக்கையில் அதிவேக செயல்பாடு கொண்டது மற்றும் ஒற்றை ஹாஷை செயலாக்கம் செய்வதன் மூலம் இதை சரிபார்க்க முடியும்.

எமது நேர முத்திரை (டைம்ஸ்டாம்ப்) நெட்வொர்க்கிற்கு, பிளாக்கின் ஹாஷுக்கு தேவையான பூஜ்ஜிய பிட்களை வழங்கும் மதிப்பு கண்டறியப்படும் வரை, பிளாக்கில் ஒரு நோன்ஸை (ஒரு முறை பயன்படுத்தப்பட்ட தற்போக்கு எண்) அதிகரிப்பதன் மூலம் பணிச்சான்றை நாங்கள் செயல்படுத்துவோம். ஒருமுறை பணிச்சான்றை திருப்திப்படுத்த CPU முயற்சி மேற்கொள்ளப்பட்ட பிறகு, பணியை மீண்டும் செய்யாமல் தொகுதியை (பிளாக்கை) மாற்ற முடியாது. பிந்தைய பிளாக்குகள் அதன் பிறகு சங்கிலியால் பிணைக்கப்படுவதால், பிளாக்கை மாற்றுவதற்கான பணியில் அதன் பிறகு அனைத்து தொகுதிகளையும் மீண்டும் செய்வது உள்ளடங்கும்.



பணிக்கான சான்று, பெரும்பான்மை முடிவெடுப்பதில் பிரதிநிதித்துவத்தை தீர்மானிப்பதில் உள்ள சிக்கலையும் தீர்க்கிறது. பெரும்பான்மையானவர்கள் ஒரு ஐபி-முகவரி-ஒரு வாக்கு அடிப்படையில் இருந்தால், பல ஐபிகளை ஒதுக்கீடு செய்யக்கூடிய எவராலும் அதை நிலைகுலையச் செய்யலாம். பணிக்கான சான்று என்பது அடிப்படையில் ஒரு-CPU-ஒரு வாக்கு என்பதைக் குறிக்கிறது. பெரும்பான்மை முடிவு மிக நீளமான சங்கிலியால் பிரதிநிதித்துவம் செய்யப்படுகிறது; அதில் முதலீடு செய்யப்பட்ட பணிக்கான மிகப்பெரிய ஆதாரம் உள்ளது. பெரும்பாலான CPU ஆற்றலானது நேர்மையான கணுக்களால் கட்டுப்படுத்தப்பட்டால், நேர்மையான சங்கிலியானது வேகமாக வளர்ந்து எந்த போட்டிச் சங்கிலிகளையும் விஞ்சும். கடந்தகால பிளாக்கை மாற்ற, தாக்குபவர், பிளாக் மற்றும் அதற்குப் பின் உள்ள அனைத்துத் தொகுதிகளின் பணிச் சான்றிதழை தாக்குதல் நடத்துபவர் மீண்டும் செய்தாக வேண்டும்; அதன் பின்னர் நேர்மையான கணுக்களின் பணியை விஞ்ச வேண்டும். அடுத்தடுத்த பிளாக்குகள் சேர்க்கப்படும்போது, மெதுவாகத் தாக்குபவர் அதை எட்டிப்பிடிக்கும் நிகழ்வு அதிவேகமாகக் குறைகிறது என்பதை நாங்கள் பின்னர் வெளிப்படுத்துவோம்.

அதிகரிக்கும் வன்பொருள் வேகம் மற்றும் காலப்போக்கில் கணுக்களை இயக்குவதில் உள்ள மாறுபடும் ஆர்வத்திற்கு ஈடுசெய்ய, பணிச்சான்றுக்கான சிரமமானது, ஒரு மணி நேரத்திற்கு பிளாக்குகளின் சராசரி எண்ணிக்கையை இலக்காகக் கொண்ட நகரும் சராசரி மூலம் தீர்மானிக்கப்படுகிறது. அவை மிக வேகமாக உருவாக்கப்பட்டால், சிரமமும் அதிகரிக்கிறது

5. நெட்வொர்க்

நெட்வொர்க்கை இயக்குவதற்கான படிநிலைகள் பின்வருமாறு:

1. புதிய பரிவர்த்தனைகள் அனைத்து கணுக்களுக்கும் ஒலிபரப்பப்படும்.
2. ஒவ்வொரு கணுவும் புதிய பரிவர்த்தனைகளை ஒரு பிளாக்கிற்குள் சேகரிக்கிறது.
3. ஒவ்வொரு கணுவும் அதன் பிளாக்கிற்கான கடினமான பணிச்சான்றை கண்டுபிடிப்பது மீது பணியாற்றுகிறது.
4. ஒரு கணு, பணிக்கான சான்றை கண்டறியும்போது, அந்த பிளாக்கை எல்லா கணுக்களுக்கும் அது ஒலிபரப்புகிறது.

5. அதில் உள்ள அனைத்து பரிவர்த்தனைகளும் செல்லத்தக்கதாகவும் மற்றும் ஏற்கனவே செலவழிக்கப்படாதவாறும் இருந்தால் மட்டுமே கணுக்கள் அந்த பிளாக்கை ஏற்றுக்கொள்ளும்.
6. முந்தைய ஹாஷாக ஏற்றுக்கொள்ளப்பட்ட பிளாக்கின் ஹாஷைப் பயன்படுத்தி, சங்கிலியில் அடுத்த பிளாக்கை உருவாக்கும் பணியை செய்வதன் மூலம் பிளாக்கை ஏற்றுக்கொண்டதை கணுக்கள் வெளிப்படுத்துகின்றன

கணுக்கள் எப்போதும் மிக நீளமான சங்கிலியை சரியான ஒன்றாக கருதுகின்றன மற்றும் அதை நீட்டிப்பதில் தொடர்ந்து செயல்படுகின்றன. இரண்டு கணுக்கள் அடுத்த பிளாக்கின் வெவ்வேறு பதிப்புகளை ஒரே நேரத்தில் ஒளிபரப்பினால், சில கணுக்கள் ஒன்று அல்லது மற்றொன்றை முதலில் பெறலாம். அத்தகைய நேர்வில், அவர்கள் முதலில் பெற்றதன் மீது பணி செய்கிறார்கள். ஆனால் அது நீண்டதாக மாறுமானால், மற்ற கிளையை சேமிப்பார்கள். பணிக்கான அடுத்த சான்று கிடைத்தவுடன், ஒரு கிளை நீளமாகும்போது இணைப்பு / கட்டு உடைக்கப்படும்; மற்ற கிளையில் பணிபுரியும் கணுக்கள் பின்னர் நீண்டதாக இருப்பதற்கு மாறிக்கொள்ளுங்கள்.

புதிய பரிவர்த்தனை ஒலிபரப்புகள் எல்லா கணுக்களையும் சென்றடைய வேண்டிய அவசியமில்லை. அவை பல கணுக்களை சென்றடையும் வரை, அவை நீண்ட காலத்திற்கு முன்பே ஒரு பிளாக்கிற்குள் வந்துவிடும். பிளாக் ஒலிபரப்புகள் கைவிடப்பட்ட செய்திகளையும் பொறுத்துக்கொள்ளும். ஒரு கணு ஒரு பிளாக்கைப் பெறவில்லை என்றால், அது அடுத்த தொகுதியைப் பெறும்போது அதைக் கோரும் மற்றும் ஒன்றை தான் தவறிவிட்டதை உணரும்.

7. **ஊக்கத்தொகை**

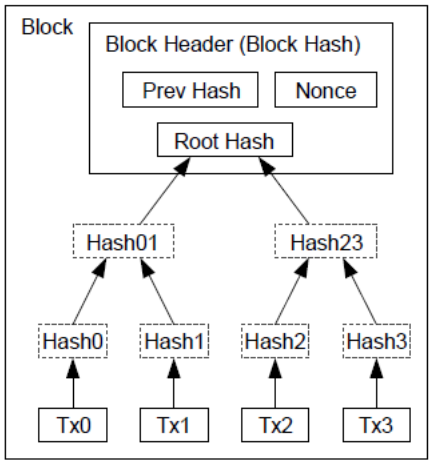
மரபுப்படி, ஒரு தொகுதியில் முதல் பரிவர்த்தனை என்பது தொகுதியை உருவாக்கியவருக்குச் சொந்தமான ஒரு புதிய நாணயத்தைத் தொடங்கும் சிறப்புப் பரிவர்த்தனையாகும். இது நெர்வொர்க்கை ஆதரிக்க கணுக்களுக்கு ஊக்கமளிக்கிறது. மேலும் நாணயங்களை வெளியிடுவதற்கு மத்திய அதிகார அமைப்பு எதுவும் இல்லாததால், ஆரம்பத்தில் அவற்றை புழக்கத்தில் விநியோகிக்க ஒரு வழிமுறையை வழங்குகிறது. புதிய நாணயங்களின் அளவை தொடர்ந்து நிலையாக சேர்த்துக்கொண்டே போவது என்பது, தங்கச் சுரங்கத் தொழிலாளர்கள் தங்கத்தை புழக்கத்தில் சேர்க்க ஆதார வளங்களைச் செலவழிப்பதைப் போன்றது. நமது விஷயத்தில், இது CPU நேரம் மற்றும் செலவழிக்கப்படும் மின்சாரம் என்பவையாக இருக்கின்றன.

பரிவர்த்தனை கட்டணத்தைக் கொண்டு ஊக்கத்தொகைக்கு நிதியளிக்கப்படலாம். ஒரு பரிவர்த்தனையின் வெளியீட்டு மதிப்பு அதன் உள்ளீட்டு மதிப்பை விட குறைவாக இருந்தால், வித்தியாசம் என்பது பரிவர்த்தனையை உள்ளடக்கிய பிளாக்கின் ஊக்கத்தொகை மதிப்போடு சேர்க்கப்படும் பரிவர்த்தனை கட்டணமாகும். முன்னரே தீர்மானிக்கப்பட்ட எண்ணிக்கையிலான நாணயங்கள் புழக்கத்தில் நுழைந்தவுடன், ஊக்கத்தொகை முற்றிலும் பரிவர்த்தனை கட்டணமாக மாறலாம் மற்றும் முற்றிலும் பணவீக்கமின்றி இருக்கும்.

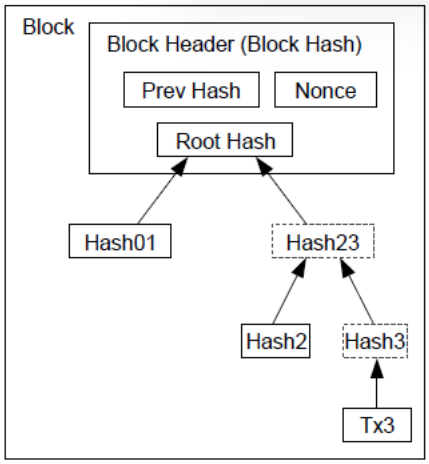
ஊக்கத்தொகை கணுக்களை நேர்மையாக செயல்படுமாறு ஊக்குவிக்க உதவும். ஒரு பேராசை கொண்ட தாக்குதல் நடத்துபவர் அனைத்து நேர்மையான கணுக்களையும் விட அதிகமான CPU சக்தியைச் சேகரிக்க முடிந்தால், அவரது பேமெண்ட்களை திருடுவதன் வழியாக மக்களை ஏமாற்றுவதற்கு அதனை பயன்படுத்துவது அல்லது புதிய நாணயங்களை உருவாக்க அதைப் பயன்படுத்துவது என்ற இரண்டில் ஒன்றை தேர்வுசெய்ய வேண்டும். அவர் தனது சொந்த செல்வத்தின் செல்லுபடித்தன்மையையும், அமைப்பையும் குறைமதிப்பிற்கு உட்படுத்துவதை விட, பிற அனைத்து நபர்களையும் ஒருங்கிணைத்ததை விட அதிக புதிய நாணயங்களை அவருக்கு வழங்கி அவருக்கு சாதகமாக இருக்கின்ற விதிகளின்படி செயல்படுவதை மிகவும் இலாபகரமானதாக அவர் காணவேண்டும்.

8. **டிஸ்க் இடத்தை மீட்டெடுப்பது**

ஒரு நாணயத்தில் சமீபத்திய பரிவர்த்தனை போதுமான பிளாக்குகளின் கீழ் புதையுண்ட உடன், டிஸ்க் இடத்தை சேமிக்க அதற்கு முன் செலவழிக்கப்பட்ட பரிவர்த்தனைகளை விலக்கிவிடலாம். பிளாக்கின் ஹாஷை உடைக்காமல் இதை ஏதுவாக்குவதற்கு, பரிவர்த்தனைகள் மெர்க்கல் மரத்தில் ஹேஷ் செய்யப்படுகின்றன [7][2][5]. மேலும் தொகுதியின் ஹாஷில் வேர் மட்டுமே சேர்க்கப்படுகிறது. மரத்தின் கிளைகளைத் துண்டிப்பதன் மூலம் பழைய பிளாக்குகளை சுருக்கி கச்சிதமாக்கலாம். உட்புற ஹாஷ்களை சேமிக்க தேவையில்லை.



Transactions Hashed in a Merkle Tree

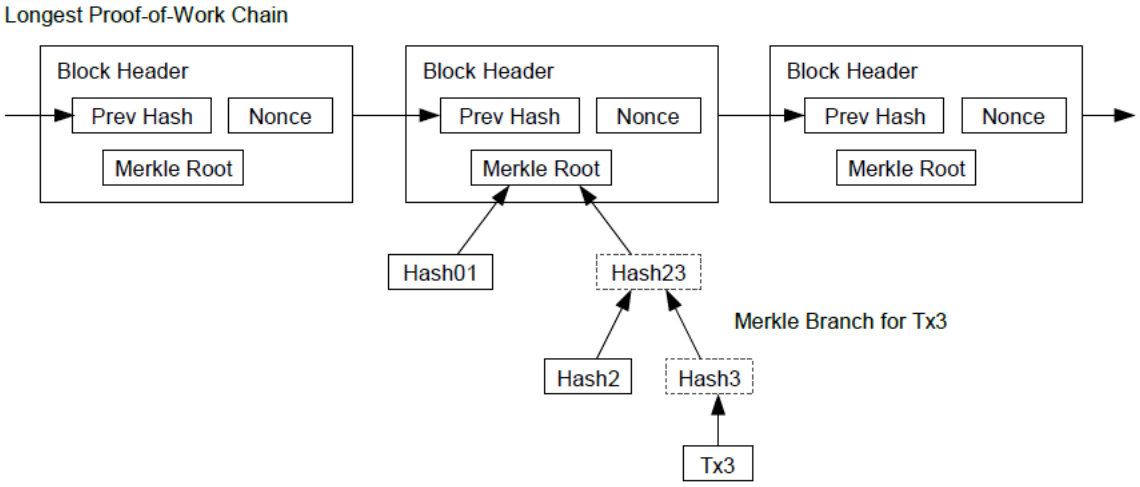


After Pruning Tx0-2 from the Block

பரிவர்த்தனைகள் இல்லாத ஒரு பிளாக்கின் தலைப்பு சுமார் 80 பைட்டுகளாக இருக்கும். ஒவ்வொரு 10 நிமிடங்களுக்கும் பிளாக்குகள் உருவாக்கப்படும் என்று நாம் யூகித்தால், ஒரு ஆண்டிற்கு 80 பைட்டுகள் * 6 * 24 * 365 = 4.2MB என்பதாக அது இருக்கும். கணினி சாதனங்கள் பொதுவாக 2008 ஆம் ஆண்டில், 2ஜிபி ரேம் உடன் விற்பனையாகின. மேலும் மூர் - ன் கோட்பாடு வருடத்திற்கு 1.2ஜிபி என்று தற்போதைய வளர்ச்சியைக் கணித்திருப்பதால், பிளாக் தலைப்புகளை மெமரியில் வைத்திருக்க வேண்டியிருந்தாலும் சேமிப்பு ஒரு சிக்கலாக இருக்காது.

9. எளிமைப்படுத்தப்பட்ட பேமெண்ட் சரிபார்ப்பு

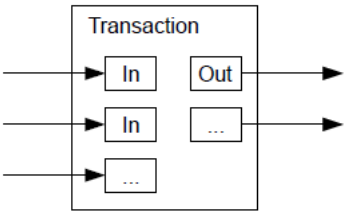
முழு நெட்வொர்க் கணுவை இயக்காமல் பணம் செலுத்தல்களை (பேமெண்ட்) சரிபார்த்து உறுதிசெய்வது சாத்தியம். நேர முத்திரையிடப்பட்டுள்ள பிளாக் உடன் பரிவர்த்தனையை பிணைக்கின்ற Merkle கிளையைப் பெற இயலும் மற்றும் தனக்கு மிக நீளமான சங்கிலி இருப்பதாக நம்பினால் தவிர, ஒரு பயனர், மிக நீளமான ப்ரூஃப்-ஆஃப்-வொர்க் சங்கிலியின் பிளாக் தலைப்புகளின் நகலை மட்டுமே வைத்திருப்பது போதுமானது. நெட்வொர்க் கணுக்களை வினவுவதன் மூலம் இதை அவர் பெற முடியும். அவரால் பரிவர்த்தனையை தனக்காகவே சரிபார்க்க முடியாது, ஆனால் சங்கிலியில் உள்ள ஒரு இடத்துடன் அதை இணைப்பதன் மூலம், ஒரு நெட்வொர்க் கணு அதை ஏற்றுக்கொண்டதையும், அதன் பிறகு பிளாக்குகள் சேர்க்கப்பட்டதையும் அவரால் பார்க்க முடியும். மேலதிக பிளாக்குகளை நெட்வொர்க் ஏற்றுக்கொண்டிருப்பதை இது மேலும் உறுதிப்படுத்துகிறது.



எனவே, நேர்மையான கணுக்கள் நெட்வொர்க்கை கட்டுப்படுத்தும் வரை சரிபார்த்து உறுதிசெய்தல் நம்பகமானதாக இருக்கும்; ஆனால் தாக்கும் நபரால் நெட்வொர்க் அடக்கி ஆட்கொள்ளப்படுமானால், அது மிகவும் பாதிக்கப்படக்கூடியதாக இருக்கும். நெட்வொர்க் கணுக்கள் தாங்களே பரிவர்த்தனைகளைச் சரிபார்க்க முடியும் என்றாலும், எளிமைப்படுத்தப்பட்ட வழிமுறையானது, தாக்குதல் நடத்தும் நபரால் நெட்வொர்க் அடக்கி ஆட்கொள்ளப்படும் காலம் வரை, அவரின் புனையப்பட்ட பரிவர்த்தனைகளால் ஏமாற்றப்படலாம்; தவறான பிளாக்கைக் கண்டறியும் போது, நெட்வொர்க் நோட்களில் இருந்து விழிப்பூட்டல்களை ஏற்றுக்கொள்வது, பயனரின் மென்பொருளை முழுபிளாக்கையும் பதிவிறக்கம் செய்யுமாறு நினைவூட்டுவது மற்றும் முரண்பாட்டை உறுதிசெய்யும் வகையில் எச்சரிக்கப்பட்ட பரிவர்த்தனைகள் ஆகியவை இதற்கு எதிராகப் பாதுகாப்பதற்கான ஒரு உத்தியாக இருக்கும். அடிக்கடி பேமெண்ட்களைப் பெறும் வணிக நிறுவனங்கள் இன்னும் சுயாதீனமான பாதுகாப்பு மற்றும் விரைவான சரிபார்ப்புக்காக தங்கள் சொந்த கணுக்களை தாங்களே இயக்க விரும்பலாம்.

10. மதிப்பை இணைத்தல் மற்றும் பிரித்தல்

நாணயங்களைத் தனித்தனியாகக் கையாள்வது சாத்தியம் என்றாலும், பரிமாற்றத்தில் ஒவ்வொரு சென்ட்டிற்கும் தனித்தனியான பரிவர்த்தனை செய்வது சிரமமாக இருக்கும். மதிப்பைப் பிரிக்கவும் மற்றும் ஒருங்கிணைக்கவும் பரிவர்த்தனைகளில் பல உள்ளீடுகள் மற்றும் வெளியீடுகள் உள்ளன. பொதுவாக, ஒரு பெரிய முந்தைய பரிவர்த்தனையிலிருந்து

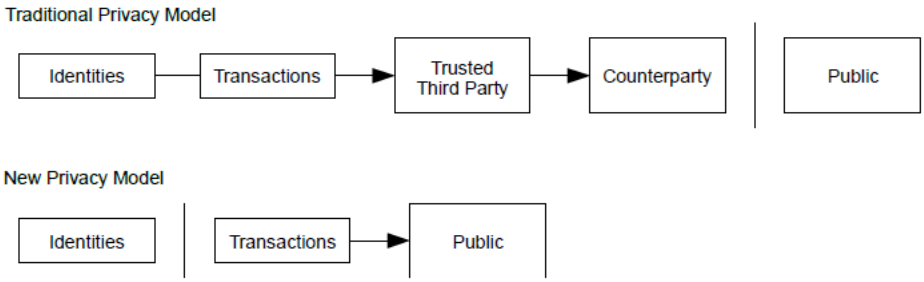


ஒரு உள்ளீடு அல்லது சிறிய தொகைகளை இணைக்கும் பல உள்ளீடுகள் இருக்கும், மேலும் அதிகபட்சம் இரண்டு வெளியீடுகள் இருக்கும். ஒன்று பணம் செலுத்துவதற்கும் மற்றொன்று திருப்பிக் கொடுக்க வேண்டிய சில்லறை இருக்குமானால், அனுப்புநருக்கு அதை திருப்பி தருவதற்குமானது.

ஒரு பரிவர்த்தனை பல பரிவர்த்தனைகளைச் சார்ந்து, அந்த பரிவர்த்தனைகள் இன்னும் பலவற்றைச் சார்ந்திருக்கும் ஃபேன்-அவுட் என்ற பரவல் செயல்பாடு இங்கு ஒரு பிரச்சனையல்ல என்பதை கவனத்தில் கொள்ள வேண்டும். பரிவர்த்தனை வரலாற்றின் முழுமையான முழுமையான நகலை பிரித்தெடுக்க வேண்டிய அவசியம் ஒருபோதும் இருக்காது.

11. தனியுரிமை (அந்தரங்க காப்பு)

பாரம்பரிய வங்கி மாதிரியானது, தொடர்புடைய தரப்பினருக்கும் நம்பகமான மூன்றாம் தரப்பிற்கும் தகவல் அணுகுவசதியைக் கட்டுப்படுத்துவதன் மூலம் அந்தரங்க காப்பை அடைகிறது. எல்லா பரிவர்த்தனைகளையும் பகிரங்கமாக அறிவிக்க வேண்டிய அவசியம் இந்த வழிமுறையைத் தடுக்கிறது, ஆனால் தகவல்களின் ஓட்டத்தை வேறொரு இடத்தில் உடைப்பதன் மூலம் தனியுரிமையை / அந்தரங்கத்தை இன்னும் பாதுகாக்க முடியும்: பொது விசைகளை (கீ) அநாமதேயமாக வைத்திருப்பதன் மூலம். யாரோ ஒருவருக்கு ஒரு தொகையை ஒருவர் அனுப்பும் பரிவர்த்தனையை யாருடனும் இணைத்துப் பார்க்கும் தகவல் இல்லாமல் மட்டுமே பொதுமக்கள் பார்க்க முடியும்; தனிப்பட்ட வர்த்தகம் மேற்கொள்ளப்பட்ட நேரம் மற்றும் அளவு குறித்த "டேப்" வெளிப்படையாகத் தெரிவிக்கப்படும் என்றாலும், அதில் ஈடுபட்ட தரப்புகள் யார் என்று தெரிவிக்கப்படாதவாறு பங்குச் சந்தைகளால் வெளியிடப்பட்ட தகவல்களின் அளவைப் போன்றதே இது.



பொதுவான உரிமையாளருடன் அவைகள் இணைக்கப்படாமல் வைக்க ஒரு கூடுதல் ஃபயர்வால் ஆக, ஒவ்வொரு பரிவர்த்தனைக்கும் ஒரு ஜோடி புதிய கீ பயன்படுத்தப்பட வேண்டும். ஒன்றுக்கும் மேற்பட்ட பல உள்ளீடு பரிவர்த்தனைகளில் சில இணைப்புகள் இந்த நிலையிலும் தவிர்க்க முடியாதவையாக, இது அவற்றின் உள்ளீடுகள் ஒரே உரிமையாளருக்குச் சொந்தமானது என்பதை அவசியம் வெளிப்படுத்துகிறது. ஆபத்து என்னவென்றால், ஒரு கீ - ன் உரிமையாளர் யார் என்று வெளிப்படுத்தப்படுமானால், அதே உரிமையாளருக்குச் சொந்தமான பிற பரிவர்த்தனைகளை லிங்கிங் / பிணைப்பு வெளிப்படுத்தக்கூடும்.

12. கணக்கீடுகள்

நேர்மையான சங்கிலியை விட மிக வேகமாக மாற்று சங்கிலியை உருவாக்க முயலும் தாக்குதல் தொடுக்கும் ஒரு காட்சியை இப்போது நாம் பரிசீலிக்கலாம். இது நிறைவேற்றப்பட்டாலும் கூட, காற்றில் இருந்து மதிப்பை உருவாக்குவது அல்லது தாக்குபவர்களுக்கு ஒருபோதும் சொந்தமாக இல்லாத பிற பணத்தை எடுப்பது போன்ற தன்னிச்சையான மாற்றங்களுக்கு இது அமைப்பைத் திறந்து விடாது. கணுக்கள், ஒரு தவறான பரிவர்த்தனையை பேமெண்ட்டாக ஏற்கப் போவதில்லை; மேலும் நேர்மையான கணுக்கள் அவற்றைக் கொண்ட ஒரு பிளாக்கை ஒருபோதும் ஏற்கமாட்டார்கள். தாக்குபவர், சமீபத்தில் செலவழித்த பணத்தைத் திரும்பப் பெறுவதற்காக தனது சொந்த பரிவர்த்தனைகளில் ஒன்றை மட்டுமே மாற்ற முயற்சிக்க முடியும்.

நேர்மையான சங்கிலிக்கும் தாக்குபவர் சங்கிலிக்கும் இடையிலான பந்தயத்தை ஒரு 'பைனோமியல் ரேண்டம் வாக்' என்று வகைப்படுத்தலாம். வெற்றி நிகழ்வு என்பது நேர்மையான சங்கிலியை ஒரு பிளாக்கால் நீட்டி, அதன் முன்னணியை +1 ஆல் அதிகரிப்பதாகும்; தோல்வி நிகழ்வு என்பது தாக்குபவர்களின் சங்கிலி ஒரு தொகுதியால் நீட்டிக்கப்பட்டு, இடைவெளியை -1 ஆல் குறைப்பதாகும்.

பற்றாக்குறை நிலையிலிருந்து அதை மாற்றி தாக்குபவர் கைவசப்படுத்துவதற்கான சாத்தியக்கூறு, சூதாட்டக்காரர்களின் அழிவு பிரச்சனைக்கு ஒப்பானது. வரம்பற்ற கிரெடிட்டைக் கொண்ட ஒரு சூதாட்டக்காரர் பற்றாக்குறை நிலையில் தொடங்கி, சமநிலையை அடைய எண்ணற்ற முயற்சிகளை நிகழ்த்துகிறார் என்று வைத்துக்கொள்வோம். அவர் எப்போதாவது சமநிலையை அடையும் சாத்தியக்கூறை நம்மால் கணக்கிட முடியும் அல்லது தாக்குதல் நடத்தும் ஒருவர் நேர்மையான சங்கிலியை

எப்போது எட்டுவார் என்று கணக்கிட முடியும். அது கீழ்வருமாறு [8]:

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

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

p > q என்ற எங்கள் அனுமானத்தின் அடிப்படையில், தாக்குபவர் கைப்பற்ற வேண்டிய பிளாக்குகளின் எண்ணிக்கை அதிகரிக்கும் போது சாத்தியக்கூறு அதிவேகமாகக் குறைகிறது. அவருக்கு எதிரான அம்சங்கள் இருக்கின்ற நிலையில்,, அவர் ஆரம்பத்திலேயே அதிர்ஷ்டவசமாக முன்னேறவில்லை என்றால், அவர் மேலும் பின்தங்கிவிடுவார் மற்றும் அவரது வாய்ப்புகள் சிறியதாகி மறைந்துவிடும்.

அனுப்பியவரால், பரிவர்த்தனையை மாற்ற இயலாது என்று நாம் போதுமான அளவு நிச்சயமாவதற்கு முன்பு வரை, புதிய பரிவர்த்தனையைப் பெறுபவர் எவ்வளவு காலம் காத்திருக்க வேண்டும் என்பதை நாம் இப்போது பரிசீலிக்கலாம். அனுப்பிய நபர் தாக்குபவர் என்றும் கருதுகிறோம். சிறிது காலத்திற்கு தான் பணம் செலுத்தியிருப்பதாக பெறுபவரை நம்ப வைக்க விரும்புகிறார் மற்றும் அதன் பிறகு சிறிது நேரம் கடந்த பிறகு தனக்கே அப்பணம் திரும்பி வருமாறு அதை மாற்றிக் கொள்கிறார் என்று நாம் யூகிக்கலாம். அது நிகழும்போது பெறுநருக்கு எச்சரிக்கை அனுப்பப்படும். ஆனால், அனுப்புபவர் அது மிக தாமதமாக இருக்கும் என்று நம்புகிறார்.

பெறுநர் ஒரு ஜோடி புதிய கீ – ஐ உருவாக்கி, கையொப்பமிடுவதற்கு சற்று முன் அனுப்புநருக்கு பொது கீயை தருகிறார். மிக அதிகமாக முன்னேறிச் செல்லும் அளவிற்கு அதிர்ஷ்டசாலியாக இருந்து, அந்நேரத்தில் பரிவர்த்தனையை செயலாக்கம் செய்தாலொழிய, தொடர்ச்சியாக அதன்மீது பணியாற்றுவதன் மூலம் முன்கூட்டியே பிளாக்குகளின் சங்கிலியை தயார் செய்வதிலிருந்து அனுப்புநரை இது தடுக்கிறது. பரிவர்த்தனை அனுப்பப்பட்டதும், நேர்மையற்ற அனுப்புநர், தனது பரிவர்த்தனையின் மாற்று பதிப்பைக் கொண்ட இணை சங்கிலியில் ரகசியமாக பணியாற்றத் தொடங்குகிறார்.

ஒரு பிளாக்கில் பரிவர்த்தனை சேர்க்கப்படும் வரை மற்றும் அதன் பிறகு z பிளாக்குகள் இணைக்கப்படும் வரை பெறுநர் காத்திருக்கிறார்; தாக்குபவர் செய்த முன்னேற்றத்தின் சரியான அளவு என்னவென்று அவருக்குத் தெரியாது; ஆனால் நேர்மையான பிளாக்குகள், ஒரு பிளாக்கிற்கு சராசரியாக எதிர்பார்க்கப்படும் நேரத்தை எடுத்துக் கொண்டன என்று யூகித்தால், தாக்குபவர்களின் சாத்தியமான முன்னேற்றம் எதிர்பார்க்கப்படும் மதிப்புடன் Poisson விநியோகமாக இருக்கும்:

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

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
```

13. **முடிவுரை**

நம்பிக்கையை சார்ந்திராமல் மின்னணு பரிவர்த்தனைகளுக்கான ஒரு அமைப்பு முறையை நாங்கள் முன்மொழிந்துள்ளோம். உரிமைத்துவத்தின் மீது உறுதியாக கட்டுப்பாட்டை வழங்குகின்ற, டிஜிட்டல் கையொப்பங்களிலிருந்து தயாரிக்கப்பட்ட நாணயங்களின் வழக்கமான கட்டமைப்பில் நாங்கள் தொடங்கினோம்; ஆனால் இரட்டைச் செலவினத்தைத் தடுக்க ஒரு வழிமுறை இல்லாமல் இது பூர்த்தியடையாது. இதைத் தீர்க்க, நேர்மையான கணுக்கள் CPU சக்தியின் பெரும்பகுதியைக் கட்டுப்படுத்தினால், தாக்குபவருக்கு மாற்றுவதற்கு விரைவாக கணக்கீட்டு ரீதியாக நடைமுறைக்கு சாத்தியமற்ற பரிவர்த்தனைகளின் பொது வரலாற்றைப் பதிவுசெய்ய, பணிச் சான்றுகளைப் பயன்படுத்தி பியர்-டு-பியர் நெட்வொர்க்கை நாங்கள் முன்மொழிந்தோம். நெட்வொர்க், அதன் கட்டமைக்கப்படாத எளிமையில் வலுவானது. கணுக்கள், சிறிதளவு ஒருங்கிணைப்புடன் அனைத்தும் ஒரே நேரத்தில் பணியாற்றுகின்றன. எந்த குறிப்பிட்ட இடத்திற்கும் செய்திகள் அனுப்பப்படாமல், சிறந்த முயற்சியின் அடிப்படையில் மட்டுமே வழங்கப்பட வேண்டும் என்பதால், அவற்றை அடையாளம் காண வேண்டிய அவசியமில்லை. கணுக்கள், அவர்கள் வெளியேறி சென்ற காலத்தில் என்ன நடந்தது என்பதற்கான ஆதாரமாக பணிச்சான்று சங்கிலியை ஏற்றுக்கொண்டு, விருப்பப்படி நெட்வொர்க்கை விட்டு வெளியேறலாம் மற்றும் மீண்டும் சேரலாம். அவர்கள் தங்கள் CPU சக்தியுடன் வாக்களிக்கிறார்கள்; செல்லுபடியாகும் பிளாக்குகளை நீட்டிப்பதன் மூலமும்,

தவறான பிளாக்குகள் மீது பணியாற்ற மறுப்பதன் மூலமும் செல்லத்தக்க பிளாக்குகளின் மீதான தங்கள் ஏற்பை வெளிப்படுத்துகிறார்கள். இந்த ஒருமித்த இயக்க முறையின் மூலம் தேவைப்படுகின்ற எந்த விதிகளும் ஊக்குவிப்பு திட்டங்களும் செயல்படுத்தப்படலாம்.

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.