Bitcoin: 'n Eweknie-elektroniese Kontantstelsel

deur Satoshi Nakamoto 2008/10/31

Abstrak

'n Suiwer eweknie-weergawe van elektroniese kontant sal toelaat dat aanlynbetalings direk van een party na 'n ander gestuur word sonder om deur 'n finansiële instelling te gaan. Digitale handtekeninge bied 'n deel van die oplossing, maar die belangrikste voordele gaan verlore as 'n betroubare derde party steeds vereis word om dubbelbesteding te voorkom. Ons stel 'n oplossing vir die dubbelbestedingsprobleem voor deur 'n eweknienetwerk te gebruik. Die netwerk stempel transaksies met tyd deur dit in 'n deurlopende ketting van huts-gebaseerde bewys-van-werk te huts, wat 'n rekord vorm wat nie verander kan word sonder om die bewys-van-werk oor te doen nie. Die langste ketting dien nie net as bewys van die volgorde van gebeure wat waargeneem word nie, maar bewys dat dit van die grootste poel van SVE-krag afkomstig is. Solank as wat 'n meerderheid van die SVE-krag beheer word deur nodusse wat nie saamwerk om die netwerk aan te val nie, sal hulle die langste ketting genereer en aanvallers verbysteek. Die netwerk self vereis minimale struktuur. Boodskappe word op 'n beste poging-basis uitgesaai, en nodusse kan na willekeur die netwerk verlaat en weer by aansluit, en die langste bewys-van-werk-ketting as bewys van wat gebeur het aanvaar terwyl hulle weg was.

Inleiding

Handel op die internet het byna uitsluitlik staatgemaak op finansiële instellings wat as betroubare derde partye dien om elektroniese betalings te verwerk. Alhoewel die stelsel goed genoeg werk vir die meeste transaksies, ly dit steeds onder die inherente swakhede van die vertrouensgebaseerde model. Heeltemal nie-omkeerbare transaksies is nie werklik moontlik nie, aangesien finansiële instellings nie bemiddeling van geskille kan vermy nie. Die koste van bemiddeling verhoog transaksiekoste, beperk die minimum praktiese transaksiegrootte en sny die moontlikheid vir klein toevallige transaksies af, en daar is 'n groter koste in die verlies aan vermoë om nie-omkeerbare betalings vir nie-omkeerbare dienste te maak. Met die moontlikheid van omkering, versprei die behoefte aan vertroue. Handelaars moet versigtig wees vir hul klante en hulle vir meer inligting vra as wat hulle andersins sou nodig hê. 'n Sekere persentasie bedrog word as onvermydelik aanvaar. Hierdie koste en betalingsonsekerhede kan persoonlik vermy word deur fisiese geldeenheid te gebruik, maar geen meganisme bestaan om betalings oor 'n kommunikasiekanaal te maak sonder 'n betroubare party nie.

Wat nodig is, is 'n elektroniese betalingstelsel gebaseer op kriptografiese bewys in plaas van

vertroue, wat enige twee gewillige partye toelaat om direk met mekaar transaksies te doen sonder dat 'n betroubare derde party nodig is. Transaksies wat rekenaarmatig onprakties is om om te keer, sal verkopers teen bedrog beskerm, en roetine-escrow-meganismes kan maklik geïmplementeer word om kopers te beskerm. In hierdie artikel stel ons 'n oplossing vir die dubbelbestedingsprobleem voor deur 'n eweknie-verspreide tydstempelbediener te gebruik om berekeningsbewyse van die chronologiese volgorde van transaksies te genereer. Die stelsel is veilig solank eerlike nodusse gesamentlik meer SVE-krag beheer as enige samewerkende groep aanvallernodusse.

Transaksies

Ons definieer 'n elektroniese muntstuk as 'n ketting van digitale handtekeninge. Elke eienaar dra die muntstuk na die volgende oor deur 'n huts van die vorige transaksie en die publieke sleutel van die volgende eienaar digitaal te onderteken en dit aan die einde van die munt te voeg. 'n Begunstigde kan die handtekeninge verifieer om die eienaarskapsketting te verifieer.

Die probleem is natuurlik dat die begunstigde nie kan verifieer dat een van die eienaars nie die muntstuk dubbel bestee het nie. 'n Algemene oplossing is om 'n betroubare sentrale gesag, of munt, in te stel wat elke transaksie vir dubbelbesteding kontroleer. Na elke transaksie moet die muntstuk na die munt terugbesorg word om 'n nuwe munt uit te reik, en slegs munte wat direk van die munt uitgereik word, word vertrou om nie dubbel bestee te word nie. Die probleem met hierdie oplossing is dat die lot van die hele geldstelsel afhang van die maatskappy wat die munt bestuur, met elke transaksie wat deur hulle moet gaan, net soos 'n bank.

Ons het 'n manier nodig vir die begunstigde om te weet dat die vorige eienaars geen vroeëre transaksies onderteken het nie. Vir ons doeleindes is die vroegste transaksie die een wat tel, so ons gee nie om oor latere pogings om dubbel te bestee nie. Die enigste manier om die afwesigheid van 'n transaksie te bevestig, is om bewus te wees van alle transaksies. In die muntgebaseerde model was die munt bewus van alle transaksies en het besluit watter eerste aangekom het. Om dit sonder 'n betroubare party te bewerkstellig, moet transaksies in die openbaar aangekondig word[1], en ons het 'n stelsel nodig vir deelnemers om ooreen te kom oor 'n enkele geskiedenis van die volgorde waarin dit ontvang is. Die begunstigde benodig bewys dat die meerderheid nodusse ten tyde van elke transaksie ooreengekom het dat dit die eerste is wat ontvang is.

Tydstempelbediener

Die oplossing wat ons voorstel, begin met 'n tydstempelbediener. 'n Tydstempelbediener werk deur 'n huts van 'n blok items te neem om 'n tydstempel te kry en die huts wyd te publiseer, soos in 'n koerant of 'n Usenet-plasing[2-5]. Die tydstempel bewys dat die data op daardie tydstip moes bestaan het, natuurlik, om in die huts te kom. Elke tydstempel sluit die vorige tydstempel in sy huts

in, wat 'n ketting vorm, met elke bykomende tydstempel wat die voor dit versterk.

Bewys-van-werk

Om 'n verspreide tydstempelbediener op 'n eweknie-basis te implementeer, sal ons 'n bewys-van-werk-stelsel soortgelyk aan Adam Back se Hashcash[6] moet gebruik, eerder as koerant- of Usenet-plasings. Die bewys-van-werk behels die skandering vir 'n waarde dat wanneer gehuts, soos met SHA-256, die huts met 'n aantal nul bits begin. Die gemiddelde werk benodig is eksponensieel in die aantal nul bits wat benodig word en kan geverifieer word deur 'n enkele huts uit te voer.

Vir ons tydstempelnetwerk implementeer ons die bewys-van-werk deur 'n nonce in die blok te verhoog totdat 'n waarde gevind word wat die blok se huts die vereiste nul bits gee. Sodra die SVE-poging bestee is om dit aan die bewys-van-werk te maak, kan die blok nie verander word sonder om die werk oor te doen nie. Aangesien latere blokke daarna vasgeketting word, sal die werk om die blok te verander insluit om al die blokke daarna oor te doen.

Die bewys-van-werk los ook die probleem op om verteenwoordiging in meerderheidsbesluitneming te bepaal. As die meerderheid gebaseer was op een-IP-adres-een-stem, kan dit ondermyn word deur enigiemand wat baie IP's kan toeken. Bewys-van-werk is in wese een-SVE-een-stem. Die meerderheidsbesluit word verteenwoordig deur die langste ketting, wat die grootste bewys-van-werk-poging daarin belê het. As 'n meerderheid van SVE-krag deur eerlike nodusse beheer word, sal die eerlike ketting die vinnigste groei en enige mededingende kettings verbysteek. Om 'n vorige blok te wysig, sal 'n aanvaller die bewys-van-werk van die blok en alle blokke daarna moet oordoen en dan die werk van die eerlike nodusse inhaal en oortref. Ons sal later wys dat die waarskynlikheid dat 'n stadiger aanvaller inhaal eksponensieel afneem soos wat daaropvolgende blokke bygevoeg word.

Om te vergoed vir die verhoging van hardeware spoed en wisselende belangstelling in lopende nodusse oor tyd, word die bewys-van-werk moeilikheid bepaal deur 'n bewegende gemiddelde gerig op 'n gemiddelde aantal blokke per uur. As hulle te vinnig gegenereer word, neem die moeilikheidsgraad toe.

Netwerk

Die stappe om die netwerk te laat loop is soos volg:

1. Nuwe transaksies word na alle nodusse uitgesaai.
2. Elke nodus versamel nuwe transaksies in 'n blok.

3. Elke nodus werk daaraan om 'n moeilike bewys-van-werk vir sy blok te vind.
4. Wanneer 'n nodus 'n bewys-van-werk vind, saai dit die blok uit na alle nodusse.
5. Nodusse aanvaar die blok slegs as alle transaksies daarin geldig is en nie reeds bestee is nie.
6. Nodusse spreek hul aanvaarding van die blok uit deur te werk aan die skep van die volgende blok in die ketting, deur die huts van die aanvaarde blok as die vorige huts te gebruik.

Nodusse beskou altyd die langste ketting as die korrekte een en sal aanhou werk om dit uit te brei. As twee nodusse verskillende weergawes van die volgende blok gelyktydig uitsaai, kan sommige nodusse die een of die ander eerste ontvang. In daardie geval werk hulle aan die eerste een wat hulle ontvang het, maar bêre die ander tak vir ingeval dit langer word. Die verbinding sal gebreek word wanneer die volgende bewys-van-werk gevind word en een tak langer word; die nodusse wat op die ander tak gewerk het, sal dan oorskakel na die langer een.

Nuwe transaksie-uitsendings hoef nie noodwendig alle nodusse te bereik nie. Solank hulle baie nodusse bereik, sal hulle binnekort in 'n blok kom. Blokuitsendings is ook verdraagsaam teenoor afneemende boodskappe. As 'n nodus nie 'n blok ontvang nie, sal dit dit versoek wanneer dit die volgende blok ontvang en besef dat dit een gemis het.

Aansporing

Volgens konvensie is die eerste transaksie in 'n blok 'n spesiale transaksie wat 'n nuwe munt begin wat besit word deur die skepper van die blok. Dit voeg 'n aansporing vir nodusse by om die netwerk te ondersteun, en bied 'n manier om munte aanvanklik in sirkulasie te versprei, aangesien daar geen sentrale gesag is om dit uit te reik nie. Die konstante toevoeging van 'n konstante hoeveelheid nuwe munte is analoog aan goudmyners wat hulpbronne bestee om goud by sirkulasie te voeg. In ons geval is dit SVE tyd en elektrisiteit wat bestee word.

Die aansporing kan ook met transaksiefooie gefinansier word. As die uitsetwaarde van 'n transaksie minder as die insetwaarde daarvan is, is die verskil 'n transaksiefooi wat by die aansporingswaarde van die blok wat die transaksie bevat gevoeg word. Sodra 'n voorafbepaalde aantal munte sirkulasie betree het, kan die aansporing geheel en al na transaksiefooie oorgaan en heeltemal inflasievry wees.

Die aansporing kan help om nodusse aan te moedig om eerlik te bly. As 'n hebsugtig aanvaller meer SVE-krag as al die eerlike nodusse kan bymekaarmaak, sal hy moet kies tussen die gebruik daarvan om mense te bedrieg deur sy betalings terug te steel, of om dit te gebruik om nuwe munte te genereer. Hy behoort dit meer winsgewend te vind om volgens die reëls te speel, sulke reëls wat hom bevoordeel met meer nuwe munte as almal saam, as om die stelsel en die geldigheid van sy eie rykdom te ondermyn.

Herwinning van Skyfspasie

Sodra die jongste transaksie in 'n muntstuk onder genoeg blokke begrawe is, kan die bestee transaksies voor dit mee weggedoen word om skyfspasie te bespaar. Om dit te fasiliteer sonder om die blok se huts te breek, word transaksies gehuts in 'n Merkle-boom [7][2][5], met slegs die wortel ingesluit in die blok se huts. Ou blokke kan dan gekompakteer word deur takke van die boom af te stomp. Die interieur hutse hoef nie gestoor te word nie.

'n Blokopskrif sonder transaksies sal ongeveer 80 grepe wees. As ons veronderstel dat blokke elke 10 minute gegenereer word, 80 grepe * 6 * 24 * 365 = 4.2MB per jaar. Met rekenaarstelsels wat tipies verkoop word met 2GB RAM vanaf 2008, en Moore se Wet wat huidige groei van 1.2GB per jaar voorspel, behoort berging nie 'n probleem te wees nie, selfs al moet die blokopskrifte in die geheue gehou word.

Vereenvoudigde Betalingsverifikasie

Dit is moontlik om betalings te verifieer sonder om 'n volledige netwerknodus te laat loop. 'n Gebruiker hoef net 'n afskrif van die blokopskrifte van die langste bewys-van-werk-ketting te hou, wat hy kan kry deur netwerknodusse te bevraagteken totdat hy oortuig is dat hy die langste ketting het, en die Merkle-tak kry wat die transaksie aan die blok koppel waarbinne dit getydstempel is. Hy kan nie self die transaksie kontroleer nie, maar deur dit aan 'n plek in die ketting te koppel, kan hy sien dat 'n netwerknodus dit aanvaar het, en blokke wat daarna bygevoeg word, bevestig verder dat die netwerk dit aanvaar het.

As sodanig is die verifikasie betroubaar solank eerlike nodusse die netwerk beheer, maar is meer kwesbaar as die netwerk deur 'n aanvaller oorweldig word. Terwyl netwerknodusse self transaksies kan verifieer, kan die vereenvoudigde metode geflous word deur 'n aanvaller se vervaardigde transaksies solank as wat die aanvaller kan voortgaan om die netwerk te oorweldig. Een strategie om hierteen te beskerm, sal wees om waarskuwings van netwerknodusse te aanvaar wanneer hulle 'n ongeldige blok opspoor, wat die gebruiker se sagteware aanpor om die volledige blok en gewaarskude transaksies af te laai om die teenstrydigheid te bevestig. Besighede wat gereelde betalings ontvang, sal waarskynlik steeds hul eie nodusse wil bestuur vir meer onafhanklike sekuriteit en vinniger verifikasie.

Kombinering en Verdeling van Waarde

Alhoewel dit moontlik sou wees om munte individueel te hanteer, sal dit moeilik wees om 'n aparte transaksie vir elke sent in 'n oordrag te maak. Om waarde toe te laat om verdeel en gekombineer te word, bevat transaksies veelvuldige insette en uitsette. Normaalweg sal daar óf 'n enkele inset van 'n groter vorige transaksie óf veelvuldige insette wees wat kleiner bedrae kombineer, en hoogstens twee uitsette: een vir die betaling, en een wat die verandering, indien enige, terugstuur

aan die sender.

Daar moet kennis geneem word dat uitwaaier, waar 'n transaksie afhanklik is van verskeie transaksies, en daardie transaksies van baie meer afhang, nie 'n probleem hier is nie. Daar is nooit die behoefte om 'n volledige selfstandige kopie van 'n transaksie se geskiedenis te onttrek nie.

Privaatheid

Die tradisionele bankmodel bereik 'n vlak van privaatheid deur toegang tot inligting tot die betrokke partye en die vertroude derde party te beperk. Die noodsaaklikheid om alle transaksies in die openbaar aan te kondig verhinder hierdie metode, maar privaatheid kan steeds gehandhaaf word deur die vloei van inligting op 'n ander plek te breek: deur publieke sleutels anoniem te hou. Die publiek kan sien dat iemand 'n bedrag aan iemand anders stuur, maar sonder inligting wat die transaksie aan enigiemand koppel. Dit is soortgelyk aan die vlak van inligting wat deur aandelebeurse vrygestel word, waar die tyd en grootte van individuele transaksies, die "band", openbaar gemaak word, maar sonder om te sê wie die partye was.

As 'n bykomende brandmuur moet 'n nuwe sleutelpaar vir elke transaksie gebruik word om te verhoed dat hulle aan 'n gemeenskaplike eienaar gekoppel word. Sommige koppeling is steeds onvermydelik met multi-insettransaksies, wat noodwendig openbaar dat hul insette deur dieselfde eienaar besit word. Die risiko is dat indien die eienaar van 'n sleutel geopenbaar word, dat die koppeling aan ander transaksies kan openbaar wat aan dieselfde eienaar behoort het.

Berekeninge

Ons beskou die scenario van 'n aanvaller wat probeer om 'n alternatiewe ketting vinniger as die eerlike ketting te genereer. Selfs al word dit bereik, gooi dit nie die stelsel oop vir arbitrêre veranderinge nie, soos om waarde uit die lug te skep of geld te vat wat nooit aan die aanvaller behoort het nie. Nodusse gaan nie 'n ongeldige transaksie as betaling aanvaar nie, en eerlike nodusse sal nooit 'n blok aanvaar wat dit bevat nie. ’n Aanvaller kan net probeer om een van sy eie transaksies te verander om geld terug te neem wat hy onlangs bestee het.

Die wedloop tussen die eerlike ketting en 'n aanvallerketting kan gekenmerk word as 'n Binomiale Lukrake Stap. Die suksesgebeurtenis is die eerlike ketting wat met een blok verleng word, wat sy voorsprong met +1 vergroot, en die mislukkingsgebeurtenis is die aanvaller se ketting wat met een blok verleng word, wat die gaping met -1 verminder.

Die waarskynlikheid dat 'n aanvaller 'n gegewe tekort sal inhaal, is analoog aan 'n Gambler's Ruin-probleem. Gestel 'n dobbelaar met onbeperkte krediet begin by 'n tekort en speel moontlik 'n oneindige aantal proewe om gelykbreekpunt te probeer bereik. Ons kan die waarskynlikheid dat hy

ooit gelykbreekpunt bereik, of dat 'n aanvaller ooit die eerlike ketting inhaal, soos volg bereken[8]:

p = waarskynlikheid dat 'n eerlike nodus die volgende blok vind
q = waarskynlikheid dat die aanvaller die volgende blok vind
q_z = waarskynlikheid dat die aanvaller ooit van z-blokke z agter sal inhaal

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

Gegewe ons aanname dat $p > q$, die waarskynlikheid eksponensieel afneem namate die aantal blokke wat die aanvaller moet inhaal, toeneem. Met die kans teen hom, as hy nie vroeg 'n gelukkige sprong vorentoe maak nie, word sy kanse verdwynend klein soos hy verder agter raak.

Ons oorweeg nou hoe lank die ontvanger van 'n nuwe transaksie moet wag voordat hy voldoende seker is dat die sender nie die transaksie kan verander nie. Ons neem aan die sender is 'n aanvaller wat die ontvanger vir 'n rukkie wil laat glo dat hy hom betaal het, en dit dan oorskakel om na 'n rukkie aan homself terug te betaal. Die ontvanger sal gewaarsku word wanneer dit gebeur, maar die sender hoop dit sal te laat wees.

Die ontvanger genereer 'n nuwe sleutelpaar en gee die publieke sleutel aan die sender kort voor ondertekening. Dit verhoed dat die sender 'n ketting blokke voor die tyd voorberei deur voortdurend daaraan te werk totdat hy gelukkig genoeg is om ver genoeg vooruit te kom, en dan die transaksie op daardie oomblik uit te voer. Sodra die transaksie gestuur is, begin die oneerlike sender in die geheim werk aan 'n parallelle ketting wat 'n alternatiewe weergawe van sy transaksie bevat.

Die ontvanger wag totdat die transaksie by 'n blok gevoeg is en z-blokke is daarna gekoppel. Hy weet nie die presiese hoeveelheid vordering wat die aanvaller gemaak het nie, maar met die aanname dat die eerlike blokke die gemiddelde verwagte tyd per blok geneem het, sal die aanvaller se potensiële vordering 'n Poisson-verspreiding met verwagte waarde wees:

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

Om die waarskynlikheid te kry dat die aanvaller nou nog kan inhaal, vermenigvuldig ons die Poisson-digtheid vir elke hoeveelheid vordering wat hy kon gemaak het met die waarskynlikheid dat hy van daardie punt af kon inhaal:

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

Herrangskik om te verhoed dat die oneindige stert van die verspreiding opgetel word…

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

Skakel tans oor na C-kode …

```
#include
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;
}
```

As ons 'n paar resultate uitvoer, kan ons sien dat die waarskynlikheid eksponensieel afneem met 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
```

Oplossing vir P minder as 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
```

Gevolgtrekking

Ons het 'n stelsel vir elektroniese transaksies voorgestel sonder om op vertroue staat te maak. Ons het begin met die gewone raamwerk van munte gemaak van digitale handtekeninge, wat sterk beheer oor eienaarskap bied, maar onvolledig is sonder 'n manier om dubbelbesteding te voorkom. Om dit op te los, het ons 'n eweknienetwerk voorgestel wat bewys-van-werk gebruik om 'n publieke geskiedenis van transaksies op te teken wat vinnig rekenaarmatig onprakties word vir 'n aanvaller om te verander as eerlike nodusse 'n meerderheid van SVE-krag beheer. Die netwerk is robuust in sy ongestruktureerde eenvoudigheid. Nodusse werk op een slag met min koördinasie. Hulle hoef nie geïdentifiseer te word nie, aangesien boodskappe nie na enige spesifieke plek herlei word nie en slegs op 'n beste poging afgelewer moet word. Nodusse kan na willekeur die netwerk verlaat en weer by aansluit, en die bewys-van-werk-ketting aanvaar as bewys van wat gebeur het terwyl hulle weg was. Hulle stem met hul SVE-krag, spreek hul aanvaarding van geldige blokke uit deur daaraan te werk om dit uit te brei en ongeldige blokke te verwerp deur te weier om daaraan te werk. Enige nodige reëls en aansporings kan met hierdie konsensusmeganisme afgedwing word.

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.