Bitcoin:

Diru elektronikorako “peer-to-peer” sistema

Satoshi Nakamoto
www.bitcoin.org

Laburpena. Alderdi berdinkide edo parekoen artean online ordainketak zuzenean, hau da, finantza erakunde baten bitartekaritzarik gabe, egiteko aukera eman beharko luke (*peer-to-peer)* berdinkidetik-berdinkiderako edo parekoen arteko sisteman guztiz oinarrituko litzatekeen diru elektronikoak. Soluzioaren parte ditugu sinadura elektronikoak, baina onura nagusiak galdu egiten dira baldin eta konfidantzazko hirugarren bat ezinbestekoa bada gastu bikoitzari aurre egin behar zaion unean. Konponbide bat proposatu dugu gastu bikoitzak dakarren arazoaren aurrean: *parekoen arteko sarea.*

Sareak denbora-zigilu baten bitartez markatzen du transakzio bakoitza, *hash* jakin batekin kodeturik, eta hashean oinarrituriko lanaren-froga diren aldetik, lanaren-froga berriro egin ezean, aldaezina izango den erregistro katea sortuko da.

Honenbestez, sortzen den kate luzeena, gertaeren ordena sekuentzialaren lekukotza izateaz gain, PUZ (prozesu unitate zentral) baliabide erreserba indartsuenetik datorrenaren froga ere izango da. PUZ baliabide gehiena, sarea erasotzeko kooperatzen ez duten nodoek kontrolatzen duten heinean, hauek sortuko dute kate luzeena erasotzaileak atzean utziz. Sareak berak, gutxieneko egitura bat baino ez du behar, mezuak era egokienean zabaltzen dira eta nodoek noiznahi egin dezakete bat sarearekin edo utzi, (*proof-of-work)* lanaren-froga kate luzeenari men eginez beti ere, ez zeuden bitartean gertatu denaren froga adierazten duenez gero.

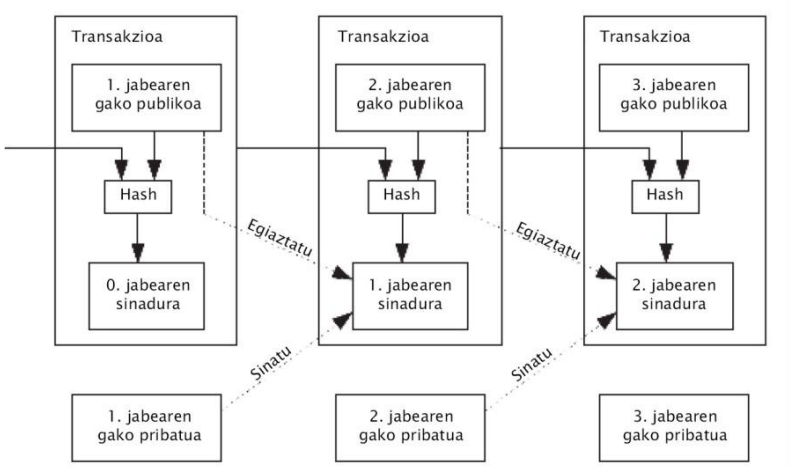
1. Sarrera

Interneteko merkaritza, ia guztiz, ondorengoan oinarritzen da: ordainketa elektronikoak prozesatzeko bitartekaritza egingo duen konfidantzazko hirugarren alderdiaren rola jokatuko duten finantza erakundeen beharrizanean. Sistema horrek transakzio gehienak egiteko nahiko ondo funtzionatzen duen arren, konfidantzan oinarrituriko ereduei datxekien ahultasunak agerian ditu, inolaz ere. Guztiz itzulerarik gabeko transakzioak ez dira posible, finantza erakundeek ezin baitute bitartekari izateari utzi desadostasunen bat gertatzen denean. Bitartekaritza horren kostuak transakzioaren kostuak igotzen ditu, haien gutxieneko neurri baliagarria mugatuz eta unean uneko transakzio txikiak gauzatzeko aukera baztertuz, eta kostu are handiagoa sorrarazten du itzuliezinezko ordainketak egiteko gaitasuna ez izateak itzuliezinezko zerbitzuen aurrean. Transakzioa itzuligarria izateko aukera horren aurrean, konfidantza beharra areagotu egiten da. Merkatariek gero eta kontu handiagoz jokatu beharra dute beren bezeroen aurrean, eta bestela beharko ez liratekeen datuak eskatzen dizkiete. Transakzio guztien gaineko ehuneko kopuru jakin baten iruzurra jasatea, saihestu ezinezkotzat onartzen da. Kostu eta ordainketen gaineko ziurgabetasuna zuzenean eginez eta eskudiru fisikoa erabiliz ekidin daiteke, baina ez dago mekanismorik ordainketak komunikazio-kanal baten bitartez egiteko konfidantzazko bitartekaririk gabe.

Konfidantzan oinarritu ordez, frogapen kriptografikoan oinarrituz, konfidantzazko hirugarren alderdi baten beharrik gabe, beraien borondatez nahi duten pareko alderdi bik elkarrekin zuzenean transakzioa egitea ahalbideratzen duen ordainketa elektronikoko sistema bat da behar dena. Informatikoki itzularazteko traketsak diren transakzioek saltzaileak iruzurretik babestuko lituzkete, eta ohiko fidantza mekanismoak erreztasunez jarriko lirateke abian erosleak babesteko. Transakzioak informatikoki itzuli ezin badira, saltzaileak iruzurren arriskutik kanpo geldituko dira eta erraz hitzartu ahal izango da eroslea babestuko duen bermearen depositurako bideren bat. Dokumentu honetan *gastu-bikoitzaren* arazoari konponbidea proposatzen diogu, pareko alderdien sare elkarbanatuan denbora-zigilu (*timestamp*) zerbitzari bat erabiliz, transakzioen ordena kronologikoaren froga informatikoa sortzeko. Nodo zintzoek nodo erasotzaile kooperatzaileen batura baino PUZ potentzia gehiago kontrolatzen duten bitartean, sistema segurua izango da.

2. Transakzioak

Txanpon elektroniko bat, sinadura digitalen kate gisa definitzen dugu. Jabe bakoitzak hurrengoari txanpona transferituko dio aurreko transakzioaren hasha eta hurrengo jabearen giltz publikoa digitalki sinatuz eta hau txanponaren amaieran gehituz. Hartzaileak sinadurak egiazta ditzake, jabetza katea ere egiaztatuz.

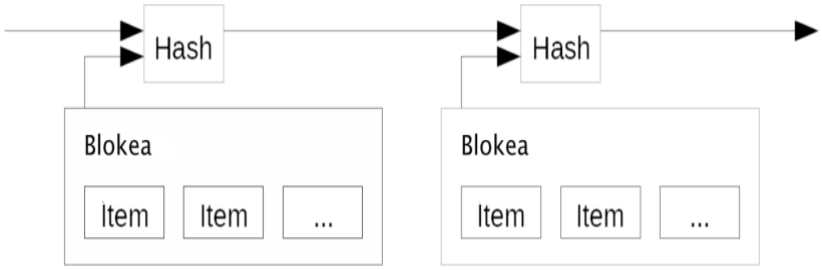


Kontua da ordaintzaileak ezin duela egiaztatu jabeetako batek ez duela bi aldiz gastatu txanpon berbera. Ohiko irtenbidea konfidantzazko autoritate zentral batengana jotzea da, transakzio guztiak ikuskatu ditzan gastu bikoitzik ez dela egon ziurtatzeko. Transakzio bakoitzaren ondoren, txanpona diru-etxera itzuli behar da txanpon berri bat egiteko, eta diru-etxetik zuzenean ateratzen diren txanponek bakarrik izango dute bi aldiz gastatu ez diren bermea. Konponbide horren arazoa zera da, diru sistema osoaren patua, diru-etxea kudeatzen duenaren esku dagoela, eta transakzio bakoitza haien bidez egin behar dela, banku batean bezalaxe.

Hartzaileak aurreko jabeek beste transakziorik sinatu ez izana guztiz ziurtatuko duen sistema behar da, inolaz ere. Gure helburua lortzeko, azken transakzioa da kontuan hartu beharrekoa, ez baikara ondoko gastu bikoitzaz arduratu beharko. Transakziorik ez dela egin egiaztatzeko modu bakarra, transakzio guztiak ezagutzea da. Diru-etxearen ereduaren arabera, ansakzio guztien berri duen diru-etxeak berak erabakitzen du zeintzuk izan ziren heltzean lehenak. Hori bera konfidantzazko alde baten parte hartzerik gabe gauzatu ahal izateko, publikoki iragarri behar dira transakzioak [1], eta parte hartzaileek jaso dituzten transakzioen ordenaren historia bakar eta berberarekin ados egoteko sistema bat behar dugu. Ordaina jaso duen hartzaileak transakzio bakoitza egiteko garaian, nodo gehienek lehen transakzioa izan dela onartu duten froga beharko du.

3. Denbora zigilatzeko zerbitzaria

Proposatzen dugun konponbidea denbora-zigilatzeko zerbitzari batekin hasiko da. Denbora zigilatzeko zerbitzariak zigilatu behar diren alez betetako bloke baten hasha, egunkari batean edo erabiltzaile sare [2-5] batean bailitzan bezala argitaratu eta zabalduz, ahalbideratzen du. Denbora-zigiluak frogatzen du emaniko informazioa une horretan existitzen dela, hala ez balitz, ez bailitzateke hash horretan sartuko. Denbora-zigilu bakoitzak aurreko hasharen denbora-zigilua gaineratzen du bere hashean kate bat osatuz, hurrenez hurren, gaineratzen den denbora-zigilu bakoitzak bere aurrekoak sendotuz.



4. *Proof-of-Work*

Denbora zigilatzeko, berdinkide edo parekoen arteko sare baten oinarritzen den, zerbitzari elkarbanatu bat inplementatzeko, Adam Back-en Hashcash delakoaren antzeko *proof-of-work* sistema [6] erabili beharko dugu, eta ez egunkarien edo Usenet postena. *Proof-of-work,* 'lanaren-froga' sistemaren egitekoa, SHA-256az adibidez *hash*-eatuz, hasieran zero bit kopuru jakin batekin hasiko den balio bat bilatzea da. Egin beharreko batez besteko lana esponentzialki hazten da eskatzen den zero bit kopuruaren arabera, eta hash bakarra exekutaturik egiazta daiteke.

Gure denbora zigilatzeko sarean, blokearen hashak beharko dituen zero bit kopurua, blokearen *nonce*-a *(number only used once)* ′behin bakarrik erabiltzeko zenbakia′, handiagotuz inplementatuko dugu lanaren-froga sistema, hashari behar besteko zero bit kopurua eskainiko dion balioa topatu arte. Behin PUZ-ak lanaren-froga lortu ahal izateko beharrezko lan guztia asebetetzen duenean, ezin aldatuko da blokea lana berriro egin ezean. Haren ondotik, katean bloke berriak erantsi direnez gero, bloke bat aldatu ahal izateko, haren ondotik sortu diren bloke guztiak ere berregin beharko lirateke.

Aldi berean, lanaren-froga sistemari esker, irtenbide egokia ematen zaio erabakiak hartzerakoan gehiengoa zehaztearen arazoari. Gehiengoa IP bidezko botuak kontuan harturik zehaztuko balitz, emaitza desitxuratu ahal izango luke IP asko baliatzen jakingo lukeen edonork. Lanaren-froga sistemaren arabera, funtsean, PUZ bakoitzari boto bakarra dagokio. Bloke kate luzeenak erakusten du gehiengoaren erabakia zein izan den, hartara bildu baita lanaren-froga ahalegin handiena. Konputazio indar handiena nodo zintzoen kontrolpean badago, kate zintzoa bizkorrago haziko da eta atzean utziko du aurre egin nahiko dion beste edozein kate. Iraganeko bloke bat aldatu ahal izateko, blokearen eta beraren ondoko beste bloke guztien lanaren-froga berregin beharko luke erasotzaileak, orduan gainditu ahal izango luke nodo zintzoen lana. Aurrerago erakutsiko dugu blokeak ondozka eranstearen poderioz, erasotzaile motelago bat zintzoen mailara iristeko aukerak modu esponentzialean murrizten direla. *Hardware* edo sistema informatikoen abiaduraren hazkundea eta nodo eragileek denboran barrena ageri dezaketen interes aldakorra konpentsatzeko, lanaren-frogari datxekion zailtasuna batezbesteko mugikor batek zehaztuko du, orduko, batez beste bloke kopuru jakin bat ekoizteko. Alabaina, bizkorregi sortzen badira, zailtasuna handitu egingo da.

5. Sarea

Sareak funtziona dezan urratsak honako hauek dira:

1) Nodo guztiei emango zaie transakzio berrien berri.
2) Nodo bakoitzak bloke batera bilduko ditu transakzio berriak.
3) Nodo bakoitza bere blokeari dagokion lanaren-froga zaila atzematen ahaleginduko da.
4) Nodo batek dagokion lanaren-froga atzematen duenean, blokearen berri emango die nodo guztiei.
5) Nodoek, soilik, blokea onartuko dute biltzen dituen transakzio guztiak baliozkoak badira eta aurrez gastatu ez badira.
6) Kateko hurrengo blokea sortzeko ahaleginetan hasten direnean, aurreko blokearen hasha erabiliz, onartuko dute nodoek blokea.

Nodoek beti, kate luzeena hartzen dute zuzen eta baliagarritzat eta ondoren luzatzen eta indartzen ahalegintzen dira. Bi nodok hurrengo blokeaz, aldi berean, bertsio desberdinak ematen badituzte, bertsio horietako bata edo bestea jasoko dute lehendabizikoz beste nodoek. Kasu horretan, jaso duten aurreneko bertsioa jarraituko dute lantzen, baina gorde egingo dute badaezpada ere beste adarra, luzeagoa izan daitekeen aukera alferrik ez galtzeko. Hurrengo lanaren-froga gauzatzean, hau da, hurrengo blokea topatzean hausten da berdinketa eta kateko adar bat luzeago bihurtzen da. Beste adarrean ari ziren nodoak kate luzeenean jarraituko dute lanean.

Ez da beharrezkoa nodo guztiek zuzenean transakzio berrien berri izatea. Alabaina, aski da beharrezko nodo kopuru batera iristea epe laburrean bloke baten parte izan daitezen. Blokeen zabalkundeak mezu galduak ere hartzen ditu kontuan. Nodo batek bloke berri baten berri jaso ez badu, hurrengo blokea jasotzean, bloke bat ahantzi duela ohartuko da eta galdetuko du hari buruz.

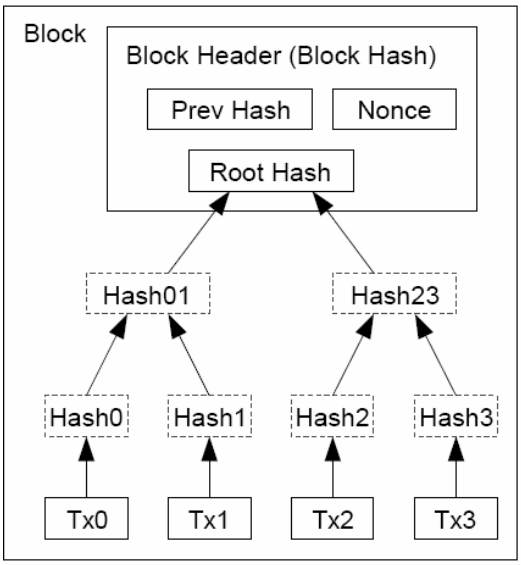
6. Pizgarria

Konbentzioz, bloke bateko lehenengo transakzioa berezia izango da jabetza izango duen txanpon berri bati hasiera ematen diolako, alegia, blokearen sortzaileari. Nodoek sarea manten dezaten, pizgarri gisa egiten du horrek, eta txanponen zirkulazioari hasiera eta banatzeko modua ematen dizkio, ez baitago txanponen emaria eskeiniko duen aginte zentralik. Txanpon berrien gehitze egonkorra, urre meatzariek urrea zirkulazioan jartzeko erabiltzen dituzten baliabide erabilpenarekin aldera daiteke. Gure kasuan, PUZ denbora eta argindarra dira erabilitako baliabideak. Transakzioen komisioen bitartez ere finantza daiteke pizgarria. Transakzioaren *outputaren* balioa, *inputaren* balioa baino txikiagoa bada, diferentzia, transakzioa jasotzen duen blokeari dagokion pizgarriaz gain, gaineratuko zaion komisiotzat hartuko da. Behin aurrez zehazturiko txanpon kopurua zirkulazioan ezarri ondoren, transakzioen komisioak izango dira beren horretan pizgarria, eta guztiz desagertuko da txanpon berrien emaria.

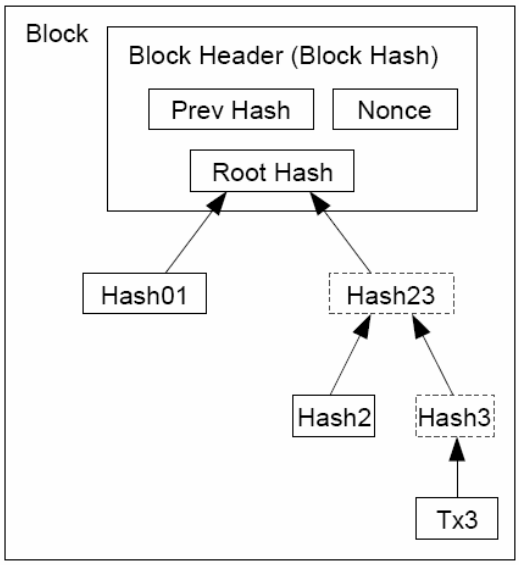
Nodoak zintzo jokatze aldera lagun dezake pizgarriak. Erasotzaile gutiziatsu batek nodo zintzo guztien konputazio indarraren batura baino gehiago biltzen badu, bi aukeren artean bat hautatu beharko du; besteei iruzur egin eta gastaturiko bere funtsak berreskuratu ala txanpon berrien emarirako erabili. Sistema eta bere ondasunen baliozkotasuna hondatzea baino, arauen bidez jokatzea onuragarriagoa iruditu beharko litzaioke, beste guztiek batera baino txanpon berri gehiago eskuratzea ahalbidaretzen baitio arau horrek.

7. Diskoan espazioa aurrezten

Txanpon baten azken transakzioa bloke kopuru baten azpitik geratzen denean, diskoan espazioa aurreztu asmoz, baztertu egin genezake. Blokearen *hasha* desegin gabe gauzatzeko, Merckle Zuhaitz eran kodetzen edo *hasheatzen* dira transakzioak [7][2][5] blokearen *root hash* edo *hash* erroa bakarrik mantenduko delarik. Ondoren, arbolaren adarrak inausiz bloke zaharrak trinkotu daitezke. Horrela, barneko *hashak* ez dira kontserbatu behar.



Merckle Zuhaitzean hasheaturiko transakzioak

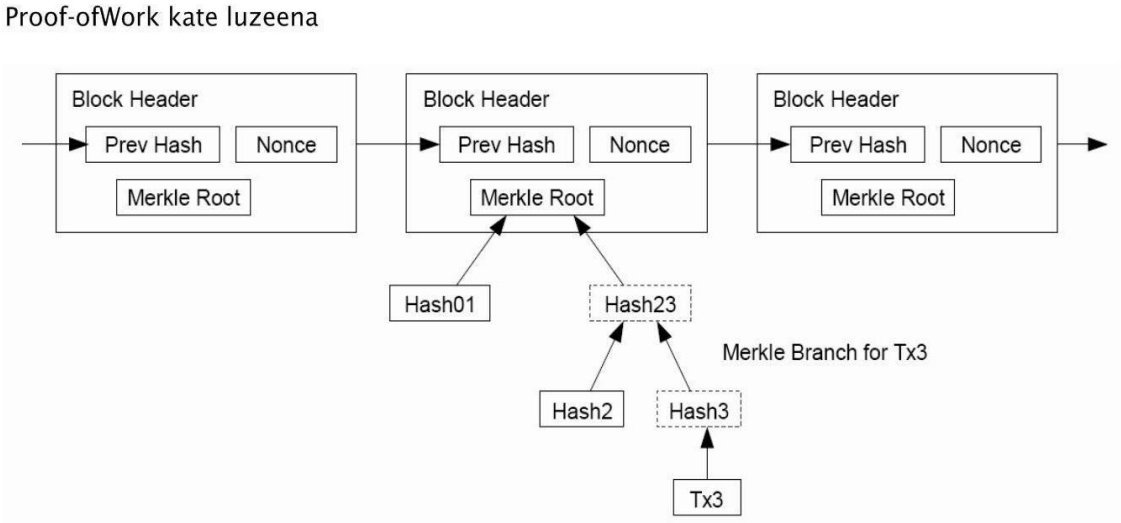


Tx0-2 Blokean inausi eta gero

Transakziorik ez daraman *block header* edo bloke buruak 80 byte inguruko pisua edukiko luke. Blokeak 10 minuturo sortzen direla jakinez gero, honako kalkulu hau egingo genuke: 80 byte * 6 * 24 * 365 = 4,2 MB urtean. 2008an merkaturaturiko ordenagailuek 2GB RAM dute batez beste, eta Mooreren Legearen arabera, urteko 1,2GBko hazkundea aurreikusi da datu biltze ahalmenaren arloan. Datu horiek guztiak kontuan harturik, espazioa ez da arazo bihurtuko *block header* edo bloke buruak memorian gorde behar badira.

8. Ordainketaren egiaztatze sinpletua

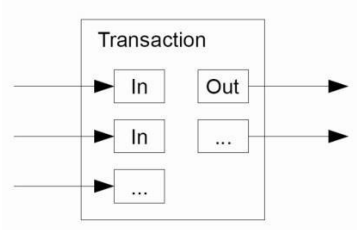
Sareko nodo osoa exekutatu beharrik gabe egiaztatu daitezke ordainketak. Horretarako, erabiltzaileak lanaren-forga kate luzeenaren *block header* edo bloke buruen kopia gordetzearekin aski izango du. Kopia hori kate luzeenarena dela ziurtatzeko, beste nodoei galdetuz egingo du, kate luzeena duela konbentzitzen den arte, denbora-zigilaturik dagoen transakzioaren blokea duen Merkle adar zuzena eskuratuz. Erabiltzaileak ezin du transakzioa zuzenean egiaztatu katean okupatzen duen lekuarekin lotuz ez bada, sareko nodoak onartu dutela ikus dezake, eta bere ondoren gaineratzen diren blokeek onarpena berresten dute.



Horrela, nodo zintzoek sarea kontrolatzen duten heinean izango da baieztapena sinesgarria, erasotzaile batek sarea menderatzen badu, baieztapenaren sinesgarritasuna ahulduz. Sareko nodoek transakzioak zuzenean egiaztatu ditzaketen arren, transakzioen metodo sinpletuari iruzur egin diezaioke sarea menderatzen duen erasotzaileak. Molde horretako erasoen aurkako babes estrategia bat, sareko nodoek balio ez duen bloke bat hauteman orduko igorririko alertak onartzea litzateke, erabiltzailearen softwarea bloke osoa eta alerta biztu dituzten transakzioak deskargatzera bultzatuz, eta funtsgabetasuna berretsiz. Ordainketak maiz jasotzen dituzten negoziek beraien nodoak erabili nahiko dituzte ziuraski, segurtasun independentzia handiagoa izateko eta egiaztapenak bizkorrago gauzatzeko.

9. Balioa Bateratzen eta Banatzen

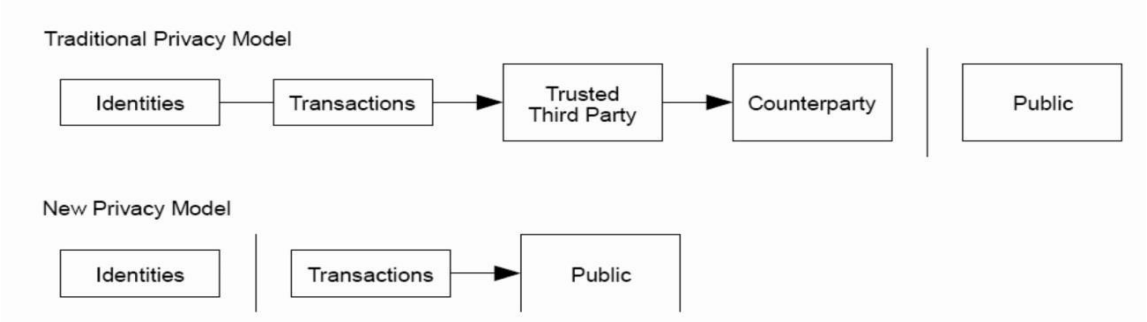
Txanponak banan-banan kudeatzea posiblea litzateken arren, ez litzateke batere praktikoa izango transferitu nahi den xentimo bakoitzeko transakazio berezi bat egitea. Balioak banatu eta batu ahal izateko, *input* eta *output* ugari hartu behar dira kontuan transakzioetan. Normalean, aurreragoko transakzio *input* edo sarrera bakar handiago bat, ala kopuru txikiagoak batzen dituzten asko egongo dira, eta gehienez *output* edo irteera bi: bat ordainketarako, eta bestea, baleude, bidaltzaileari hondarrak bueltatzeko.



Aipatu behar da, transakzio bat hainbat transakzioren mende dagoela, eta transakzio horiek beste askoren mende dauden arren, ez dagoela arazorik. Inoiz ez da transakzio baten historiaren *standalone* kopia edo kopia oso independente baten beharrik izango.

10. Pribatutasuna

Banku eredu tradizioanalak lortzen duen pribatutasun maila, informazioaren eskuragarritasuna, alderdi partehartzaileei eta konfidantzazko hirugarren alderdiari mugatuz lortzen du. Transakzio guztiak jendaurrean iragarri beharrak eragozten du jokamolde hori, baina pribatutasun maila horri eutsi ahal izateko aski da informazioaren ezagutza publikoa beste leku batean etenda: gako publikoen anonimatotasuna mantenduz. Jendeak norbaitek beste norbaiti diru kopuru bat igorri diola ikusi ahal du, baina inori lotu liezaiokeen informaziorik gabe. Burtsek eskaintzen duten informazio mailaren antzekoa da hori; izan ere, jendaurrean iragartzen dira operazioak gertatu direneko denbora eta tamaina, *"tape"* delakoa, baina alderdi partehartzaileak nortzuk diren esan gabe.



Beste *firewall* edo babeshorma gisa, gako pare berri bat baliatu beharra dago transakzio bakoitzean, jabe batekin erlazionatuak izan daitezen eragozteko. Erlazio maila bat saihestezina da transakzio input-aniztunen kasuan, ezinbestez ageriko baitute input guztien jabegoa. Gakoaren jabea nor den ezagutuz gero, jabe beraren beste transakzio batzuk agerian geratzean legoke arriskua.

11. Kalkuluak

Erasotzaile batek kate zintzoa baino bizkorrago sortaraziko lukeen ordezko kate baten hipotesia aintzat hartu behar da. Horrelakorik lorturik ere, sistema ez da aldaketa arbitrarioen arriskupean geratuko, ezingo du ezerezetik baliorik sortu edo erasotzailearena sekula izan ez den dirurik eskuratu. Nodoek ez dituzte baliorik gabeko transakzioak ordainketa gisa onartuko, eta nodo zintzoek sekula ez dute halakorik daraman blokerik onartuko. Erasotzaile bat, bakarrik, bere transakzioetako bat aldatzen saiatu daiteke gastatu berri duen dirua berreskuratzeko asmoz.

Binomial Random Walk [17] edo ausazko bide binomial baten antzera deskribatu liteke kate zintzoaren eta kate erasotzailearen arteko lasterketa. Gailentzearen arrakasta, kate zintzoa bloke batez luzatuz +1 aldearekin lidergoa lortzen duenean gertatzen da, eta porrota aldiz, erasotzaileak bere katea bloke batez luzatzea lortzean, zintzoarekiko tartea -1 murriztean.

Gambler's Ruin edo apostulariaren hondamendia problemarekin pareka liteke atzean dagoen erasotzaileak zintzoaren mailara iristeko duen probabilitatea [18]. Demagun kreditu mugagabea duen apustulari bat defizitean hasten dela, eta nahi beste proba egin ditzakela berdinketa lortzen saiatzeko. Bada, horrenbestez kalkula genezake parean gelditzeko edo, bestela esanik, erasotzaileak kate zintzoaren mailara iristeko probabilitatea [8]:

p = nodo zintzoak hurrengo blokea topatzeko probabilitatea
q = erasotzaileak hurrengo blokea topatzeko probabilitatea
q_z = erasotzailea katearen mailara iristeko probabiltatea, z blokeko desabantaila duela

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

$p > q$ kontuan hartuz, berdindu beharreko bloke kopurua handiagotzen den heinean, erasotzaileak berdintzeko dituen probabilitateak esponentzialki jausten dira. Egoera kontra duela, zorioneko bultzada goiztiar bat izan ezean, denbora igaro ahala gero eta aukera gutxiago izango ditu, gero eta atzerago jausten den erasotzaileak. Orain, transakzio berri baten hartzaileak bidaltzaileak itzuliezina duen ziurtasun onargarria izateko itxaron beharko duen denbora hausnartuko dugu. Demagun igorlea erasotzailea dela eta hartzaileari ordaindu diola sinistarazten diola une batez eta ondoren bere burari berbidaltzen diola ordainketa. Hartzaileak, horren berri jasoko duen arren, igorleak beranduegi izango den esperantza du.

Hartzaileak gako pare berri bat sortzen du eta gako publikoa ematen dio igorleari sinatu baino lehen. Jokamolde horrekin hartzaileak saihesten du igorleak denbora baino lehen beste kate bloke baten prestaketan lan egiteko aukera; horrela, kate zintzoa berdintzeko gai izan ez dadin, transkzkioa momentu horretan exekutatuz. Behin transakzioa bidalita, igorle gaiztoa sekretuan hasten da bere transakzioaren bertsio alternatibo bat duen kate paralelo batean lanean.

Transakzioa bloke batean erantsi arte eta, beraz, ondotik zeramatzan *z* blokeak ere lotu zaizkion arte, egongo da hartzailea zain. Hartzaileak ez du ezagutzen erasotzaileak egin duen aurrerakuntzaren kopuru zehatzik, baina bloke zintzoak espero zen epean sortu direla kontuan harturik, erasotzailearen balizko aurrerapena Poisson banaketaren arabera [19] espero den baliokoa izango da:

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

Erasotzaileak orain bere helburua lortzeko duen probabilitatea kalkulatzeko, honako bi faktore hauek biderkatuko ditugu: egin lezakeen aurrerapen kopuru zehatzaren Poisson dentsitatea, alde batetik, eta puntu horretatik abiaturik bloke zintzoaren mailara heltzeko duen probabilitatea, bestetik.

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

Banaketaren batuketa mugagabea saihesteko berrantolatu ondoren:

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

C kodera bihurtu eta...

```
#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;
}
```

Emaitza batzuk kontuan hartu ondoren, argi eta garbi ikus daiteke probabilitatea modu esponentzialean jausten dela z-ren kasuan:

```
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
```

Eta hona *P* % 0.1 baino txikiagoa denean:

```
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
```

12. Bukaera

Transakzio elektronikoak egiteko, konfidantzan oinarritu beharrik gabeko sistema proposatu dugu. Txanponen ohiko egituraz hasi gara, jabegoaren kontrol hertsia eskeintzen duten sinadura digitalekin abiatuz, baina gastu bikoitzaren arazoa saihesten ez duenez gero, ez da nahikoa. Hori konpontzeko berdinkidetik-berdinkiderako, parekoen sarea proposatu dugu, lanaren-froga erabiliz transakzioen historia publikoa erregistratzeko, honela, nodo zintzoek kontrolatzen badute PUZ indarraren gehiengoa, erasotzaile batentzat, gaitasun konputazionalaren ikuspuntutik eginezina izan dadin.

Sarea sendoa da bere egiturarik gabeko sinpletasunean. Nodoek batera funtzionatzen dute koordinazio gutxirekin. Ez dute identifikaziorik behar, mezuak ez baitira leku jakin batera bideratzen, baizik eta aliritzian zabaltzen. Nahi dutenean atera eta sar daitezke nodoak sarean, eta lanaren-froga katea onartu beharra dute beti, haren barnean egon ez direnean gertatu denaren froga den heinean. Beraien PUZ konputazio indarraz bozkatzen dute, lanaren bidez adierazten dute baliozko blokeen onarpena, haiek zabalduz, eta baliozkoak ez diren blokeen gaitzespena, haiek lantzea errefusatuz. Beharrezko arau eta pizgarriak kontsentsu mekanismo horrekin behartu daitezke.

Erreferentziak

[1] W. Dai, “b-money”, http://www.weidai.com/bmoney.txt, 1998.

[2] H. Massias, X.S. Avila, y J.-J. Quisquater, “Design of a secure timestamping service with minimal trust requirements”, en *20th Symposium on Information Theory in the Benelux* , 1999ko maiatza.

[3] S. Haber, W.S. Stornetta, “How to time-stamp a digital document”, en *Journal of Cryptology*, vol. 3, no. 2, 99-111 orrialdeak, 1991.

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

[5] S. Haber, W.S. Stornetta, “Secure names for bit-strings”, *Proceedings of the 4th ACM Conference on Computer and Communications Security* , 28-35 orrialdeak, 1997ko apirila.

[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”, *Proc. 1980 Symposium on Security and Privacy* , IEEE Computer Society,122 – 133 orrialdeak,1980ko apirila.

[8] W. Feller, “An introduction to probability theory and its applications”, 1957.

Dokumentu Originala

Bitcoin: A Peer-to-Peer Electronic Cash System

Itzulpena eta moldaketak
@Blooma_Lorea
Zuzenketak
SL
671.737