RaiBlocks: Satu Rangkaian Mata Wang Kripto Tanpa Bayaran

Colin LeMahieu
clemahieu@nano.co

***Abstrak*—Baru baru ini, permintaan yang tinggi dan skala yang terhad telah miningkatkan purata masa transaksi and kos bayaran dalam semua matawang kripto yang popular, menghasilkan pengalaman yang tidak memuaskan. Disini kami memperkenalkan RaiBlocks, satu matawang kripto bersenibina novel block-lattice, dimana setiap akaun mempunyai rangkaian blok tersendiri, berkemampuan untuk menghantar transaksi hampir serta merta dan berskala tanpa had. Setiap pengguna mempunyai rangkaian blok tersendiri, membolehkan mereka mengemas kini rangkaian blok tersebut secara tak serentak kepada semua rangkaian, membolehkan transaksi yang pantas dengan kos overhed yang minimal. Transaksi menjejaki baki akaun dan bukannya amaun transaksi - ini membolehkan pemangkasan pengkalan data yang aggresif tanpa menjejaskan sekuriti. Sehingga kini, rangkaian RaiBlocks telah memproses 4.2juta transaksi menggunakan rangkain lejar bersaiz hanya 1.7GB. Dengan kebolehan transaksi tanpa bayaran dan sekelip mata, ini membolehkan RaiBlock menjadi matawang kripto yang terulung.**

***Terma Indeks*—matawang kripto, blockchain, raiblocks, lejar diedarkan, digital, transaksi**

I. Introduksi

SEJAK pemgimplimentasi Bitcoin pada tahun 2009, terdapat peningkatan alihan daripada mata wang tradisional dan sistem kewangan kepada sistem pembayaran moden bersasakan kriptografi, yang berkebolehan untuk menyimpan dan memindah dana secara selamat [1]. Untuk berfungsi secara efisyen, matawang kripto perlu mudah dipindah milik, tidak boleh dibalikkan dan mempunyai bayaran yang minimum atau percuma. Kenaikan masa transaksi, bayaran yang tinggi serta skala rangkaian yang boleh dipersoalkan telah membangkitkan persoalan tentang praktikal penggunaan Bitcoin sebagai mata wang seharian.

Di dalam kertas kerja ini, kami memperkenalkan RaiBlocks, satu mata wang kripto rendah latensi dibuat atas innovasi struktur data block-lattice yang menawarkan skala tanpa had dan tiada bayaran transaksi. RaiBlocks direka bentuk dengan protokol mudah bertujuan untuk menjadi mata wang kripto berprestasi tinggi. Protokol RaiBlocks boleh dijalankan oleh perkakas berkuasa rendah, membolehkan ia menjadi praktikal, satu mata wang kripto desentralisasi kegunaan harian.

Statistik mata wang kripto dilaporkan didalam kertas kerja ini adalah tepat bersasas kan tarikh publikasi.

II. Latar Belakang

Pada tahun 2008, seorang individu tanpa nama bernama samaran Satoshi Nakamoto menerbitkan satu kertas putih menggariskan mata wang kripto desentralisasi pertama di dunia, Bitcoin [1]. Innovasi utama dibawa oleh Bitcoin adalah tentang blockchain, satu lejar awam, tidah boleh diubah dan struktur data desentralisasi yang digunakan sebagai untuk transaksi mata wang. Malangnya, dengan kematangan Bitcoin, beberapa issue didalam protokol telah membuat Bitcoin terhalang untuk kegunaan:

1) Berskala Rendah: Setiap block didalam blockchain boleh menyimpan data yang terhad, ini menyebabkan sistem hanya boleh memproses data yang tidak banyak dalam masa sesaat, membuat titik hitam didalam komoditi block. Median semasa untuk setiap bayaran transaksi adalah $10.38 [2].
2) Latensi Tinggi: Purata pengesahan transaski adalah 164 minit. [3].
3) Ketidak cekapan penggunaan kuasa: Network Bitcoin menggunakan anggaran 27.28TWh setahun, dengan purata penggunaan 260KWh setiap transaksi [4].

Bitcoin dan mata wang kripto yang lain, berfungsi dengan mencapai konsensus di dalam lejar global masing-masing untuk mengesahkan transaksi yang sah sambil menentang sumber yang berniat jahat. Bitcoin mencapai konsensus melalui model ekonomi yang dipanggil Proof of Work (PoW). Di dalam sistem PoW perserta bersaing untuk menghitung satu nombor, dipanggil *nonce*, supaya hash didalam block berada dilalam kadar sasaran. Sasaran yang sah adalah berkadar terbalik daripada kuasa pengiraan kumulatif seluruh rangkaian Bitcoin untuk mengkekalkan masa purata yang konsisten untuk mencari nonce yang sah. Pencari nonce yang sah dibernarkan untuk menambah blok kepada blockchain tersebut; Oleh itu, mereka yang menggunakan sumber komputasi yang banyak untuk mengira satu nonce memainkan peranan yang besar di dalam keadaan blockchain. PoW menyediakan rintangan terhadap serangan Sybil, dimana satu entiti berkelakuan seperti entiti perlbagai untuk meraih kuasa tambahan didalam sistem desentrilisasi, dan mengurangkan kondisi perlumbaan yang wujud secara semulajadi sambil mengakses struktur data global.

Satu alternatif protokol konsensus, Proof of Stake (PoS), pertama kalinya diperkenalkan oleh Peercoin pada tahun 2012 [5]. Di dalam sistem PoS, peserta mengundi dengan mintitik beratkan bilangan kekayaan yang mereka peroleh dalam mata wang kripto yang mereka punyai. Dengan perjanjian ini, mereka yang mempunyai pelaburan kewangan yang besar akan diberikan lebih kuasa dan secara semulajadinya deberi insentif untuk mengkekalkan ketulusan sistem atau berisiko untuk hilang pelaburan mereka. PoS dilaksanakan tanpa pembaziran kuasa pengiraan komputasi, dengan hanya memerlukan perisian ringan yang dijalankan diatas perkakas komputer berkuasa rendah.

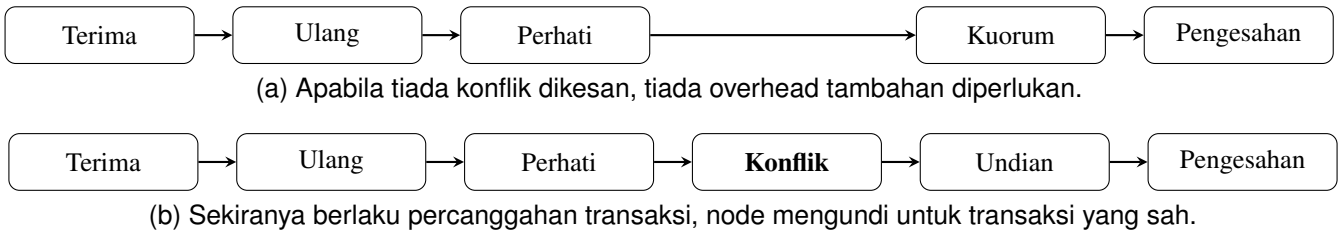

Fig. 1. RaiBlocks tidak memerlukan overhead tambahan untuk transaksi tipikal. RaiBlocks requires no additional overhead for typical transactions. Sekiranya berlaku percanggahan transaksi, node perlu mengundi untuk menyimpan transaksi.

Kertas kerja RaiBlocks asal dan implimentasi beta telah diterbitkan pada Disember, 2014, membuatkan ia antara yang pertama mata wang kripto berasaskan Directed Acyclic Graph (DAG) [6]. Tidak lama selepas itu, mata wang kripto DAG yang lain mula dibangunkan, terutamanya DagCoin/Byteball dan IOTA [7], [8]. Mata wang kripto berasakan DAG ini telah memecahkan acuan blockchain, dengnan meningkatkan prestasi sistem dan keselamatan. Byteball mencapai konsensus begantung dengan satu "main-chain" milibatkan "witnesses" yang jujur, bereputasi tinggi dan dipercayai pengguna. IOTA pula mencapai konsensus melalui PoW terkumpul oleh susunan transaksi. RaiBlocks mencapai konsensus dengan udian seimbang atas transaksi yanng bertentangan. Sistem konsensus ini menyediakan sistem yang lebih pantas, transaksi yang lebih diterministik sambil mengekalkan sistem yang kuat dan densentralisasi. RaiBlocks meneruskan pembangunan sistem ini dan di iktiraf sebagai salah satu mata wang kripto berprestasi tinggi.

III. Komponen RaiBlocks

Sebelum menggambarkan seni bina RaiBlocks secara keseluruhan, setiap komponen harus di definisikan terlebih dahulu.

A. Akaun

Akaun adalah bahagian kunci-awam (Public Key) daripada pasangan tandatangan digital kunci. Public-key, juga dirujuk sebagai alamat, di kongsi bersama perserta rangkaian tetapi kunci peribadi (Private Key) dirahsiakan. Satu paket data ditandatangani secara digital untuk memastikan kandungan adalah diluluskan oleh pemilik private-key. Satu pengguna boleh mengawal beberapa akaun, tetapi hanya satu public address boleh ada didalam satu akaun.

B. Block/Transaction

Terma "block" dan "transaction" selalu digunakan secara berganti, dimana satu block mengandungi satu transaksi. Transaksi secara khususnya merujuk kepada tindakan pengkodkan digital transaksi. Transaksi ditandatangani oleh kunci-peribadi dimiliki oleh akaun yang malakukan transaksi tersebut.

C. Ledger/ Lejar

Ledger ialah set akaun global dimana setiap akaun mempunyai transaction chain (Figure 2) tersendiri. Ini adalah komponen reka bentuk utama yang jatuh di dalam kategori menukar perjanjian jangka masa dengan perjanjian masa reka bentuk; semua pihak bersetuju memalui semakan tandatangan yang hanya boleh diubah oleh pemilik akaun dan rantai mereka sendiri. Ini mengubah satu stuktur data yang dikongsi dan lejar yang diedarkan kepada set yang tidak di kongsi.

D. Node

Satu *node* ialah satu cebisan perfisian yang dijalankan dalam komputer yang mengesahkan kenapa protokol Raiblocks dan mengambil bahagian dialam rangkaian RaiBlocks. Persirian ini mengurus lejer dan mana-mana akaun yang berada dibawah kawalan node tersebut. Node boleh menyimpan seluruh lejer atau cebisan sejarah lejer yang mengandungi beberapa block teakhir pemilik akaun. Semasa menubuhkan node baru, adalah disarankan untuk mengesahkan seluruh sejarah and memagkas secara lokal.

IV. Gambaran Sistem

Tidak seperti blockhain yang digunakan oleh mata wang kripto yang lain, RaiBlocks menggunakan satu struktur *block-lattice*. Setiap akaun mempunyai blockchain (account-chain) tersendiri sama seperti sejarak transaksi/ sejarah baki akaun (Figure 2). Setiap account-chain hanya boleh dikemas kini oleh pemilik akaun; ini membolehkan setiap account-chain dikemaskini serta-merta dan tidak segerak / asikroni kepada block-lattice yang lain, menjurus kepada transaksi serta merta.

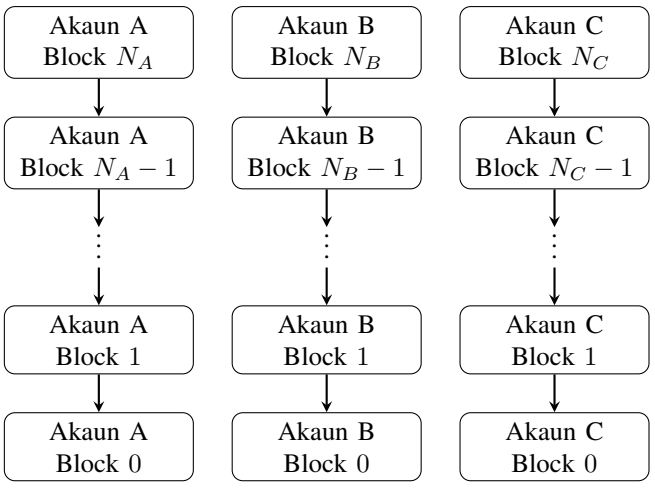

Fig. 2. Setiap akaun mempunyai blockchain tersendiri mengandungi baki akaun. Block 0 mestilah transaksi terbuka. (Section IV-B)

protokol RaiBlocks adalah sangat ringan; setiap transaksi boleh dimuatkan dalam minimum size paket UDP untuk di transmit di atas talian. Keperluan perkakasan untuk nodes adalah minimal, kerana nodes hanya diperlukan untuk rekod dan penyaran semula blocks untuk kebanyakan transaksi (Figure 1).

Sistem ini dimulakan dengan *genesis account* mengandungi *genesis balance*. Genesis balance adalah kuatiti mutlak dan tidak boleh ditambah. Genesis balance dibahagi dan dihantar kepada akaun-akaun lain melalui hantaran transaksi didaftarkan didalam genesis account-chain. Jumlah baki untuk semua akaun tidak akan melebihi jumlah asal genesis balance dimana sistem akan mendapat kelebihan kualiti dan tiada keboleh penambahan baki.

Bahagian ini akan menunjuk bagaimana perbezaan transaksi dibina dan disebarkan ke seluruh rangkaian.

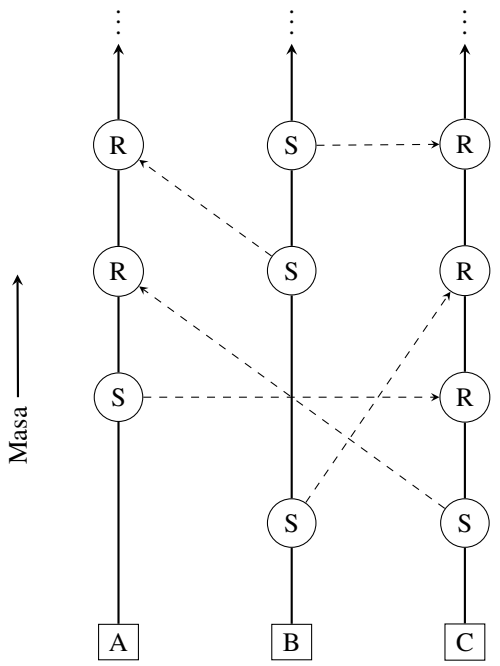

Fig. 3. Visualisai block-lattice. Setiap pemindahan dana memerlukan penghatar block (S) dan penerima block (R), dan ditandatangani oleh setiap pemilik account-chain (A,B,C)

A. Transactions

Pemindahan dana dari satu akaun ke akun lain memerlukan dua transaksi: satu *send* menolak amaun dari baki penhantar dan satu *receive* penambahan kepada baki akaun penerima(Figure 3).

Pemindahan jumlah sebagai transaksi bersasingan didalam penerima dan penghantar akaun bertujuan untuk beberapa punca utama:

1) Urutan pemindahan masuk yang semulajadi asinkroni.
2) Mengkekalkan transaksi kecil untuk diletakkan didalam paket UDP.
3) Pelancaran pengguntingan lejer melalui pengurangan jejak data.
4) Pengasingan transaksi selesai dari transaksi tidak selesai.

Operasi asinkroni adalah apabila lebih dari satu akaun memindah ke destinasi akaun yang sama; latensi rangkaian dan akaun penghantar tidak semestinnya berkomunikasi bersama, bermaksud tidak ada cara diterima universal untuk mengetahui transaksi mana yang berlaku dahulu. Kerana penambahan adalah bersekutu, perintah turutan input adalah tidak penting, dan hanya perjanjian global diperlukan. Ini adalah komponen reka bentuk utama yang menukar perjanjian jangka-masa kepada perjanjian reka bentuk-masa. Akaun penerima mempunyai kawalan terhadap pemindahan mana yang sampai dahulu dan dinyatakan melalui tandatangan kemasukan block baru.

Jika akan mahu membuat pemindahan yang besar yang diterima oleh set pemindahan yang lebih kecil, kami mahu mewakili transaksi ini untuk muat kedalam paket UDP. Apabila akaun penerima meletakkan urutan input pemindahan, ia menyimpan jumlah bergerak baki akaun tersebut supaya ia mempunyai kebolehan untuk memindah apa-apa jumpah dengan tansaksi berjumlah tepat. Ini berbeza dengan model transaksi input/output yang digunakan Bitcoin and mata wang kripto yang lain.

Sesetengan nodes tidak mempunyai keinginan untuk mengembangkan sumber bagi menyimpan senarah penuh satu akaun; mereka hanyalah berminat dengan baki didalam setiap akaun. Apabila satu akaun membuat transaksi dan node ini hanya perlu menjejaki blok terbaru, yang membolehkan mereka untuk buand jejak sejarah data sambil mengkekalkan ketepatan.

Dengan fokus untuk perjanjian reka bentuk-masa, terdapat tingkap kelewatan apabila pengesahan transaksi kerana due untuk mengenalpasti dan to identifying and pengendalian petugas tidak baik didalam rangkaian. Sejak perjanjian di RaiBloks dicapai dengan sangat cepat, dalam lingkungan millisaat ke saat, kita boleh memberi pengguna dengan dua kategori bisasa tentang transaksi: selesai and tidak selesai. Transaksi selesai adalah dimana akaun telah menghasilkan dan menerima blocks. Transaksi tidak selesai belum lagi dimasukkan kedalam akaun penerima. Ini adalah penggantian untuk condisi yang lebih kompleks and metrik pengesahan tidak dikenali di dalam mata wang kripto yang lain.

B. Pembukaan Akaun

Untuk membuat akaun, transaksi *open* perlu dikeluarkan. (Figure 4). Transaksi open adalah transaksi pertama untuk semua account-chain dan dapat dibuat dengan penerimaan dana yang pertama. Bidang *account* menyimpan alamat public-key (address)yang diambil dari private-key yang digunakan untuk ditandatangan. Bidang *source* mengandungi hash transaksi yang menghantar dana tersebut. Apabila akaun dibuka, satu wakil haruslah dipilih untuk bengundi diatas pihak pembuka akaun; ini boleh diubah kemudian masa (Section IV-F). Akaun boleh menamakan dirinya sebagai wakil sendiri.

C. Baki Akaun

Baki dalam akaun di rekod didalam lejar senndiri. Daripada merekodkan amaun transaksi, pengesahan (Section IV-I)

```
open {
   account: DC04354B1...AE8FA2661B2,
   source: DC1E2B3F7C...182A0E26B4A,
   representative: xrb_1anr...posrs,
   work: 0000000000000000,
   type: open,
   signature: 83B0...006433265C7B204
}
```

Fig. 4. Anatomi transaksi terbuka

memerlukan semakan perbezaan antara baki di block penghantaran dengan baki block sebelumnya. Akaun penerima kemudian boleh meningkatkan baki di dalam akaun seperti ukuran yang diterima di block baru. Ini adalah untun memperbaiki kelajuan pemprosesan apabila memuat turn jumlah block yang tinggi. Apabila melihat sejarah akaun, jumlah telah diberikan.

D. Penghantaran Dari Akaun

Untuk penghantaran dari alamat, alamat tersebut perlu mempunyai block yang sedia ada, dan terhasilnya baki (Figure 5). Bidang sebelum/ *previous* mengandungi hash block sebelumnya di dalam account-chain. Bidang alamat/ *destination* mengandungi akaun untuk dana yang mahu dihantarkan. Satu block penghantaran tidaklagi boleh diubah setelah mendapat pengesahan. Setelah disiarkan di dalam rangkaian, dana akan di tolah serta merta dari akaun penghantar dan ditunggu / *pending* sehingga penerima menandatangan block untuk menerima dana tersebut. Dana yang ditunggu tidah dianggap menunngu pengesahan, kerana dana tersebut boleh diambil kira sebagai telah dihantar dan transaksi tersebut tidak boleh dibatalkan.

```
send {
   previous: 1967EA355...F2F3E5BF801,
   balance: 010a8044a0...1d49289d88c,
   destination: xrb_3w...m37goeuufdp,
   work: 0000000000000000,
   type: send,
   signature: 83B0...006433265C7B204
}
```

Fig. 5. Anatomi penghantaran transaksi

E. Penerimaan Transaksi

Untuk melengkapkan transaksi, penerima haruslah membuar satu block penerimaan di dalam account-chain sendiri (Figure 6). Bidang sumber/source merujuk kepada hash transaksi yang berkenaan. Setelah block ini dibuat dan disiarkan, baki akaun akan dikemas kini dan dana telah dipindahkan ke akaun penerima.

F. Menetapkan Perwakilan

Pemilik akaun mempunyai kebolehan untuk memilih wakil untuk mengundi untuk mereka. Ini adalah alat desentralisasi

```
receive {
   previous: DC04354B1...AE8FA2661B2,
   source: DC1E2B3F7C6...182A0E26B4A,
   work: 0000000000000000,
   type: receive,
   signature: 83B0...006433265C7B204
}
```

Fig. 6. Anatomi penghantaran transaksn

yang tidak mempunyai analog yang kuat didalam protokol PoW atau PoS. Node yang sentiasa berjalan adalah tidak praktical untuk kenbanyakan pengguna; kebolehan untuk mewakilkan kuasa untuk mengundi boleh melegakan keperluan ini. Pemilik akaun mumpunyai keboleh untuk menukar konsensus kepada mana-mana akuan pada bila-bila masa. Satu perubahan/ *change* transakski mengubah perwakilan oleh satu akaun dengan menolak pemberatan undian dari wakil yang lama dan menambah pemberatan undian bagi wakil yang baru (Figure 7). Tiada dana akan dipindahkan didalam proses transaksi ini, dan pewakil tidak mempunyai kuasa untuk menggunakan dana pemilik akaun.

```
change {
   previous: DC04354B1...AE8FA2661B2,
   representative: xrb_1anrz...posrs,
   work: 0000000000000000,
   type: change,
   signature: 83B0...006433265C7B204
}
```

Fig. 7. Anatomi perubahan transaksi

G. Cabangan dan Undian

Satu cabang akan belaku apabila block yang j tandatangan $b_1, b_2, \ldots, b_j$ mendakwa blok yang sama sebagai pendahulu (Figure 8). Block ini menyebabkan pecanggahan pandangan ke atas status akaun dan ia perlu diselesaikan. Hanya pemilik akaun mempunyai kebolehan untuk menandatangani blok kepada account-chain mereka, ini meyebabkan cabang berlaku akibat programming yang lemah atau niat jahat (double-spend) oleh pemilik akaun.

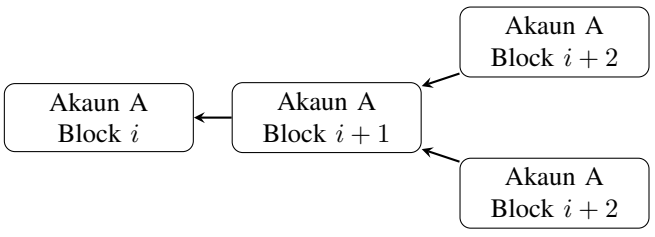

Fig. 8. Satu cabang berlaku apabila dua (atau lebih) block yang ditandatangan merujuk kepada blok terdahulu yang sama. Block lama di sebelah kiri; block baru di sebelah kanan

Apabila ia dikesan, satu wakil akan membuat satu undian yang merujuk kepada blok tersbut $\hat{b}_i$ di dalam lejarnya dan

menyiarkan ia kepada rangkaian. Pemberatan undian vote tersebut, w_i, adalah jumlah daripada semua baki akaun yang telah menamakan ia sebagai wakil mereka. Node tersebut akan memerhati semua undian yang masuk daripada semua wakil lain di dalam talian M dan akan menyimpan tally kumulatif dalam tempoh 4 undian, berjumlah 1 minit, dan mengesahkan block pemenang (Equation 1).

$$v(b_j) = \sum_{i=1}^{M} w_i \mathbb{1}_{\hat{b}_i = b_j} \tag{1}$$

$$b^* = \arg\max_{b_j} v(b_j) \tag{2}$$

Block paling popular b^* akan memperoleh undian majoriti dan akan dikekalkan sebagai lejar node tersebut (Equation 2). Block-block yang kalah akan dibuang. Jika wakil mengubah block didalam lejarnya, ia akan membuat unduan dengan turutan yang lebih tinggi dan menyiarkan undian tersebut di dalam rangkaian. **Hanya** senario inilah wakil akan mengundi.

Di dalam beberapa keaadaan, masalah sambungan kepada rangkain boleh menyebabkan blok yang tersiar untuk tidak diterima oleh semua. Block-Block seterusnya dari akaun ini akan diabaikan dan dianggap tidak sah oleh node yang tidak melihat siaran asal. Penyiaran semula akan diterima oleh node yang lain dan block seterusnya akan diterima secara automatik. Walaupun satu cabang atau blok hilang berlaku, hanya akaun yang dirujuk di dalam transaksi akan terjejas; rangkaian node yang lain akan meneruskan dengan pemprosessan akaun yang lain.

H. Proof of Work

Kesemua empat-empat jenis transaksi mempunyai satu bidang kerja yang perlu di isi dengan tepat. Bahagian kerja membolehkan tansaksi mengira satu nonce supaya hash nonce tersebut bersambung dengan dengan bahagian didalam receive/send/change sebelumnya, transaksi atau bahagian akaun di dalam satu transaksi terbuka adalah di dalam satu nilai lingkungan. Tidak seperti Bitcoin, PoW di dalam RaiBlocks hanya digunakan sebagai alat anti-spam, sama seperti Hashcash, dan boleh dikira dalam lingkungn saat [9]. Apabila transaksi telah dihantar, Pow block selepasnya boleh terdahulunya dikira kerana bahagian block sebelum telah diketahui; ini menyebabkan transaksi dilihat seperti serta-merta oleh pengguna selagi masa antara transaksi adalah lebih dari masa yang diperlukan untuk mengira PoW.

I. Verifikasi Transaksi

Untik block dikira sebagai sah, ia perlu mempunyai attribut tersebut:

1) Blok tidak boleh sudah berada di dalam lejer (Transaksi duplikat / pendua).
2) Mesti ditandatangani oleh pemilik akaun.
3) Block sebelumnya adalah kepala/ punca block account-chain. Jika ia wujud tapi bukan punca, maka ia adalah cabang.
4) Akaun harus mempunyai block terbuka.
5) Hash yang dikira haruslah memenuhi keperluan PoW.

Jika ia adalah block penerima, punca block hash perlu disemak sama ada belum selai, ini bermaksud ia belum ditebus. Jika ia adalah block penghantar, baki akaun haruslah kurang dari baki sebelumnya.

V. Vektor Penyerang

Sama seperti mata wang kripto yang lain, terdapat kebarangkalian untuk di serang oleh pihak berniat jahat untuk keuntungan kewangan atau merosakkan sistem. Di bahagian ini, kita menggariskan beberapa senario serangan yang mungkin berlaku, akibat dari hasil serangan, dan bagaimana protokol RaiBlock's mengambil langkah pencegahan.

A. Jurang Peyegerakan Block

Di dalam seksyen cabang/ IV-G, kita membincangkan tentang senario dimana satu block berkemungkinan tidak disiarkan dengan betul, menyebabkan rangkaian untuk tidak memperdulikan blok seterusnya. Jika node melihat satu blok yang tidak mempunyai rujukan block sebelumnya, ia mempunyai dua pilihan:

1) Abaikan block tersebut kerana ia berkemungkinan untuk menjadi block berniat jahat.
2) Meminta untuk di selaraskan semula dengan node lain.

Di dalam kes peyelarasan semula, satu sambungan TCP perlu dibuat dengan node 'bootstrapping' untuk memudahkan penambahan trafik yang diperlukan. Tetapi, jika block tersebut adalah block tidak sah, maka penyelarasan yang dilakukan adalah sisa-sia lalu menyebabkan penambahan trafik didalam rangkaian. Ini adalah Network Amplification Attack dan menyebabkan penafian perkhidmatan atau DDoS.

Untuk mengelak penyelarasan yang sia-sia, node-node akan menunggu sehingga satu lingukan undian telah di perhatikan yang block berniat jahat berkemungkinan wujud sebelum before sambungan kepada node bootstrap akan dimulakan. Jika undian diterima tidak mencukupi, ia boleh diambil kira sebagai data sampah.

B. Pembanjiran Transaksi

Satu entiti jahat berkemungkian menghantar transaksi sah yang banyak dan tidak diperlukan di antara akaun-akaun di bawah kawalan entiti tersebut dengan niat memenuhkan rangkaian. Serangan ini boleh dilaksanakan setiap masa disebabkan tiada kos transaksi. Walau bagaimanapun, PoW yang diperlukan oleh setiap transaksi menghadkan kadar transaksi yang boleh dilaksanakan entiti tersebut tanpa menggunakan sumber pengkiraan komputer. Malah di dalam keadaan untuk memenuhkan lejer tersebut, node yang tidak penuh boleh mencantas transaksi lama dari rantaian mereka; ini menyebabkan tiada kesempitan pengunaan simpanan dari serangan tersebut.

C. Serangan Sybil

Entiti boleh membuat beratus node-node RaiBlock di dalam satu mesin; Namun begitu, kerana sistem undian adalah berasaskan baki akaun, node-node berlebihan didalam rangkaian tidak akan menambah undian penyerang. Oleh itu, tiada kelebihan yang akan diperoleh melalui serangan Sybil.

D. Serangan Penny-Spend

Serangan penny-spend adalah dimana penyerang berbelanja menggunakan kuantiti bersaiz kecil kepada jumlah akaun yang bersar berniat untuk membazirkan suber simpanan node-node. Kadar penerbitan block adalah terhad kepada PoW, ini menghadkan pembukaan akaun dan transaksi. Node yang tidah penuh boleh mencantas akaun yang berada di bawah satu nilai dan menganggap akaun tersebut tidak sah. Akhir sekali, RaiBlocks dihasilkan supaya menggunakan rekod data kekal yang kecil, ini membolehkan simpanan yang diperlukan untuk menyimpan satu akaun lebihan adalah berkadar dengan saiz open block + indexing = 96B + 32B = 128B. Ini bermaksud 1GB boleh menyimpan sebanyak 8 juta akaun penny-spend. Jika node mahu mencantas dengan lebih aggresif, mereka boleh megira kadar pengedaran dengan kekerapan akses dan mengasingkan akaun yang jarang digunakan ke simpanan yang lebih perlahan.

E. Serangan Prakiraan/ Precomputed PoW

Oleh kerana pemilik akaun sahaja mumpunyai kebolehan untuk menambah block kepada account-chain, block seterusnya boleh dikira dengan PoWnya sekali, sebelum disiarkan kedalam rangkaian. Disini, penyerang menghasilkan pelbagai block-block seterusnya, setiap-satu mempunyai amaun yang kecil di dalam satu masa. Tiba di masa tertentu, penyerang akan melakukan penafian perkidmatan/ Denial of Service (DoS) dengan membanjiri rangkain dengan transaksi yang sah, menyebabkan node-node lain akan memproses dan menyiarkan secepat mungkin. Ini adalah versi susah daripada seksyen pembanjiran transaksi yang dinyatajan sebelum ini V-B. Penyerangan ini hanya akan berfungsi dengan sekejap, tetapi boleh digunakan bersempena dengan serangan-serangan lain, seperti >50% Attack (Section V-F) untik menigkatkan keberkesanan. Menghadkan kadar transaksi dan teknik-teknik lain sedang di selidik untuk mengurangkan serangan.

F. >50% Attack

Metrik konsesus digunakan RaiBlocks adalah mengunakan sistem undian kekuatan baki akaun. Jika penyerang boleh memperoleh lebih 50% daripada kekuatan undian, mereka boleh menyebabkan rangkian undian berolak-balik menyebabkan sistem yang rosak. Penyerang boleh mengurangkan keseimbangan undian dengan menghalang node-node baik dari mengundi melalui DoS rangkaian. RaiBlocks mengambil langkah-langkah berikut untuk mengelakkan serangan tersebut:

1) Pertahanan utama dari serangan ini adalah terikat kepada kuasa-undian terikat kepada jumlah perlaburan didalam sistem ini. Pemilik akaun semulajadinya berinsentif untuk kekalkan kejujuran sistem untuk menyelamatkan pelaburan mereka. Cubaan untuk menukar lejar akan merosakkan sistem dan memusnahkan pelaburan mereka.
2) Kos untuk melakukan serangan ini adalah berkadar dengan permodalan pemasaran RaiBlocks. Di dalam sistem PoW , teknologi mungkin boleh dihasilkan untuk memberi kuasa ketidak stabilan, berbanding pelaburan kewangan dan jika serangan ini berhasil, teknologi ini boleh diguna semula selepas serangan ini selesai. Dengan RaiBlocks kos untuk menyerang sistem bergerak dengan sistem tersebut, jika serangan itu berjaya, pelaburan oleh penyerang tidak dapat di selamatkan.
3) Untuk mengkekalkan kuorum maksima pengundi, benteng pertahanan seterusnya ialah perwakilan undian. Pemilik akaun yang tidak boleh mengundi akibat masalah sambungan ke rangkain, boleh menamakan wakil untuk mengundi berkadarkan baki bereka. Memaksimakan jumlah dan kepelbagaian menigkatkan daya tahan rangkaian.
4) Cabang di RaiBlocks tidak akan berlaku secara tidak sengaja, jadi node boleh membuat keputusan polisi atas bagaimana untuk berinteraksi dengan block cabang. Akaun bukan penyerang hanya terdedah kepada block cabangan jika mereka menerima baki dari akaun penyerang. Akaun yang mahu lebih keselamatan boleh menunggu lebih lama sebelum menerima dana dari block cabang atau menolak sama sekali. Penerima boleh juga menjana akaun berasingan untuk digunakan semamsa penerimaan dada dari akaun yang diragui untuk melindungi akaun-akaun lain.
5) Benteng pertahanan terakhir yang belum dilaksanakan adalah menyimen block/ *block cementing*. RaiBlocks berkerja keras untuk menyelesaikan cabangan block melalui undiran. Node-node boleh dikonfigurasikan untuk menyimen block, ini akan mengelakkan mereka dari diperkenalkan semula dalam suatu masa. Rangkaian ini adalah cukup selamat dengan tumpuan kepada penyelesain cepat untuk mengelakkan cabang yang disyaki.

Versi serangan $> 50\%$ yang lebih sofistikated boleh dilihat di rajah 9. “Offline” ialah peratusan perwakil yang dinamakan tetapi tidak berada di dalam talian untuk mengundi. “Stake” adalah jumlah pelaburan yang akan digunakan penyerang untuk mengundi. “Active” ialah pewakil yang berada di dalam talian dan mengundi mengikut protokol. Penyerang boleh mengofset bilangan stake / saham dengan membuat pengundi yang lain di luar talian dengan menggunakan serangan DoS ke atas rangkaian. Jika serangan ini boleh dikekalkan, pengundi-pengundi yang lain akan menjadi tidak selaras dan ini yang dinamakan “Unsync.” Akhir sekali, penyerang boleh meningkatkan kuasa undian untuk masa yang singkat dengan menukar serangan DoS/ Denial of Service ke wakil yang baru sementara set lama sedang melaraskan lejar mereka, cara ini dipanggi “Attack.”

Fig. 9. Satu serangan sususan undian yang boleh menurunkan syarat 51%.

Jika penyerang boleh membuat Stake >Active dengan gabungan keadaan berikut, mereka boleh mengubah undian di atas lejar dengan menggunakan saham kepentingan mereka. Kita boleh membuat anggaran kos untuk melakukan serangan ini dengan memeriksa kapasiti pasaran sistem lain. Jika

anggaran 33% wakil-wakil berada di luar talian atau diserang DoS, penyerang akan perlu membeli 33% daripada jumlah pasaran untuk menyerang sistem melalui undian.

G. Bootstrap Poisoning

Lebih lama penyerang boleh menyimpan private-key lama yang mempunyai baki, lebih tinggi kebarangkalian baki yang wujud pada masa itu akan mendapatthe higherana baki atau wakil mereka telah dipindahkan ke akaun baru. Ini bermaksude jika node telah di bootstrap kepada perwakilan rangkaian yang lama dimana penyerang mempunyai kuorum kuasa undian berbanding keputusan undian pada masa tersebut, mereka boleh mengolah keputusan undian node tersebut. Jika pengguna baru ini mahu berinteraksi dengan orang lain selain penyerang, semua transaksi pengguna tersebut akan ditolak kerana block utama yang berlainan. Ini menyebabkan node akan merugikan masa node-node baru didalam rangkaian disebabkan piemberian informasi yang salah. Untuk mengelakkan situasi ini, node boleh di pasangkan dengan pengkalah data akaun awal dan block utama yang diketahui sahih; ini adalah penggantian kerana memuat turun pengkalan data dari block genesis/asal. Lebih dekat proses memuat durun dengan masa semasa, lebih tinggi kebarangkalian untuk mempertahan serangan ini dengan tepat. Pada akirnya, serangan ini tidak seteruk dengan memberi data sampah kepada node yang berkeadaan bootstrap, ini kerana mereka tidak boleh berurus dengan sesiapa yang mempunyai pengkalan data kontemporari.

VI. Implimentasi

Pada masa ini, implimentasi dilaksanakan dalam C++ dan telah and menghasilkan keluaran sejak tahun 2014 di Github [10].

A. Ciri-ciri Reka Bentuk

Implimentasi RaiBlocks mengikut garis seni bina di dalam kertas kerja ini. Spesifikasi tambahan degambarkan di sini.

1) Algoritma Signing: RaiBlocks mengunakan algoritma ED25519 elliptic curve yang di ubah suai dengan Blake2b hashing untuk semua tandatangan/signatures [11]. ED25519 dipilih kerana kepantasan tandatangan/ signing, verifikasi yang laju, dan keselamatan yang kuat.

2) Algoritma Hashing: Sejak algoritma hashing digunakan unaltuk mengelak spam rangkaian, pemilihan algoritma adalah kurang penting berbanding mata wang kripto beasaskan mining. Implimentasi kami menggunakan Blake2b sebagai as a algoritma cernaan terhadap kandungan block [12].

3) Fungsi Key Derivation: Di dalam dompet rujukan, kunci dilindungi oleh kata kunci dan kata kunci tersebut di proseskan melalui fungsi key derivation untuk dilindungi oleh cubaan bukaan ASIC. Pada masa ini, Argon2 [13] merupakan pemenang pertandingan awam yang berasaskan untuk mereka satu fungsi berdaya tahan fungsi key derivation.

4) Selangan Block: Oleh kerana setiap akaun mempunyai blockchain tersendiri, proses kemas kini boleh dilakukan secara asinkroni mengikut keadaan rangkaian. Oleh itu, tiada selanganblock dan transaksi boleh disiarkan serta merta.

5) Protokol UDP Message: Sistem ini di reka bentuk untuk beroperasi selama-lamanya menggunakan sumber pengkomputeran yang minimum. Semua mesej didalam sistem di reka untuk tidak berbentuk apa-apa dan boleh dimuatkan di dalam satu paket UDP. Ini mempermudahkan golongan node yang mempunyai sambunga ynag sekejap untuk mengambil bahagian didalam rangkain tanpa perlu memulihkan sambunga TCP jangka masa pendek . TCP hanya digunakan untuk node baru apabila meraka mahu bootstrap block chains dengan banyak.

Node boleh memastikan transaksi mereka diterima rangkain dengan melihat siaran trafik transaksi dari node-node lain, kerana ia sepatutnya dapat melihat beberapa salinan dihantar kepada dirinya.

B. IPv6 dan Multicast

Penghasilan tanpa sambungan UDP membolehkan perlaksanaan penggunaan IPv6 multicast masa hadapan sebagai gantian untuk memenuhkan transaksi dan siaran undian. Ini boleh mengurangkan penggunaan jalur lebar rangkaind dan memberi lebih fleksibiliti polisi kepada node di masa hadapan.

C. Prestasi

Pada masa penulisan ini, 4.2 juta transaksi telah diproses oleh rangkain RaiBlocks, menghasilkan satu blockchain bersaiz 1.7GB. Masa transaksi didalam lingkuan berberapa saat. Rujukan boleh dilihat dengan penggunaan SSD boleh memproses lebih 10,000 transasi sesaat.

VII. Penggunaan Sumber

Ini adalah gambaran keseluruhan sumber-sumber yang digunakan node RaiBlocks. Tambahan pula, idea-idea dilihat untuk mengurangkan penggunaan sumber untuk kes-kes spesifik. Pengurangan node-node dipanggil light, cantasan/prune, atau simplified payment verification (SPV) nodes.

A. Rangkaian

Jumlah aktiviti di dalam rangkain bergantung dengan berapa banyak sumbangan oleh rangkaian terhadap kesihatan rangkaian tersebut.

1) Wakil: Satu node perwakil memerlukan sumber rangkaian maksimum kerana ia memerhati trafik undian dari wakil-wakil lain dan menyiarkan undian ia sendiri.

2) Trustless: Node trustless bersamaan dengan node wakil tetapi hanya sebagai pemerhati, ia tidak mempunyai perwakilan akaun dan tidak menghasilakan undian.

3) Trusting: Node trusting memerhadi trafik undian dari wakil yang dipercayai untik melaksanakan konsensus. Ini mengurangkan jumlah trafik undian masuk dari wakil kepada node ini.

4) Light: Sama seperti Trusting node, Node light juga hanya memerhati trafik untuk akaun yang ia berminat untuk penggunaan rangkaina yang minima.

5) Bootstrap: Satu node bootstrap berkhidmat untuk sebahagian atau semua lejar node yang baru berada di dalam talian. Ini dilaksana kan melalui sambungan TCP bukannya UDP kerana ia melibatkan jumlah data yang besar dan memerlukan flow control yang canggih.

B. Kapasiti Cakera

Bergantung dengan permintaan pengguna, node berbeza memerlukan konfigurasi yang berlainan mengukut keperluan pengguna.

1) Sejarah: Node yang berminat untuk menyimpan rekod sejarah semua transksi akan memerlukan jumlah simpanan maksima.

2) Masa kini: Oleh kerana reka bentuk simpanan baki terkumpul dengan block-block, node hanya perlu menyimpan block terbaru atau kepala block setiap akaun untuk mengambil bahagian di dalam konsensus. Jika node tidak berminat untuk menyimpan sejarah penuh lejar, ia boleh memilih untuk menyimpan kepala block sahaja.

3) Light: Node light tidak menyimpak data lejar secara lokal dan hanya mengambil bahagian di dalam rangkaina untuk memerhati akaun yang mereka minat dan diberi peluang untuk membuat transaksi baru dengan private key yang mereka simpan.

C. CPU

1) Penghasilan Transaksi: Node yang berminat untuk menghasilkan transaksi mesti membuat satu nounce Proof of Work untuk melepasi mekanisma batasan RaiBlock. Pengkiraan perkakas computer ditanda aras pada lampiran A.

2) Perwakilan: Wakil harus mengesahkan tandatangan untuk block-block, undian, dan menghalson tandatangan sendiri untuk mengambil bahagian di dalam konsensus. Amaun sumber CPU untuk node pewakil adalah jauh lebih sedikit dari penghasilan transaksi dan boleh dijalankan dengan komputer kontemporari satu CPU .

3) Pemerhati: Node permerhati tidak menghasilakan undian sendiri. Oleh kerana penghasilan tandatangan overhead adalah minima, keperluannya adalah hampir sama dengan node perwakilan.

VIII. Konklusi

Dalam kertas kerja ini, kami membentangkan rangka kerja untuk satu matawang yang dipercayai, tiada bayarann dan berlatensi rendah yang menggunakan struktur block-lattice dan and perwakilan undian Proof of Stake. Rangkaian yang memerlukan sumber minima, pekakas yang sederhana dan boleh memproses transaksi yang banyak. Semua ini boleh dicapai melalui blockchain berindividu untuk setiap akaun, mengurangkan isu-isu berlebihan dan struktur data yang tidak efisyen. Kami mengenalpasti semua kebarangkalian serangan ke atas sistem dan membentangkan semua hujan bagaimana RaiBlock boleh menahan semua bentuk serangan tersebut.

Appendix A
PoW Perkakasan Pendanda Aras

Seperti yang dinyatakan sebelum ini, PoW di dlam RaiBlocks adalah untuk mengurangkan spam rangkaian. Node implimentasi node kami membolehkan pemantasan yang boleh menggunakan kelebihan GPU-GPU berkebolehan OpenCL. Rajah I menunjukkan perbandingan antara perkakas-perkakas komputer masa kini. Buat masa ini, nilai ambang PoW adalah tetap, tetapi ambang penyesuaian mungkin di laksakan apabila kuasa pergiraan semakin berkembang.

TABLE I
Prestasi Perkakas PoW

Alat	Transaksi sesaat
Nvidia Tesla V100 (AWS)	6.4
Nvidia Tesla P100 (Google,Cloud)	4.9
Nvidia Tesla K80 (Google,Cloud)	1.64
AMD RX 470 OC	1.59
Nvidia GTX 1060 3GB	1.25
Intel Core i7 4790K AVX2	0.33
Intel Core i7 4790K,WebAssembly (Firefox)	0.14
Google Cloud 4 vCores	0.14-0.16
ARM64 server 4 cores (Scaleway)	0.05-0.07

Pengiktirafan

Kami ingin berterima kasih kepada Brian Pugh kerana menyusun dan pemformatan kertas kerja ini.

Rujukan

[1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf
[2] “Bitcoin median transaction fee historical chart.” [Online]. Available: https://bitinfocharts.com/comparison/bitcoin-median_transaction_fee.html
[3] “Bitcoin average confirmation time.” [Online]. Available: https://blockchain.info/charts/avg-confirmation-time
[4] “Bitcoin energy consumption index.” [Online]. Available: https://digiconomist.net/bitcoin-energy-consumption
[5] S. King and S. Nadal, “Ppcoin: Peer-to-peer crypto-currency with proof-of-stake,” 2012. [Online]. Available: https://peercoin.net/assets/paper/peercoin-paper.pdf
[6] C. LeMahieu, “Raiblocks distributed ledger network,” 2014.
[7] Y. Ribero and D. Raissar, “Dagcoin whitepaper,” 2015.
[8] S. Popov, “The tangle,” 2016.
[9] A. Back, “Hashcash - a denial of service counter-measure,” 2002. [Online]. Available: http://www.hashcash.org/papers/hashcash.pdf
[10] C. LeMahieu, “Raiblocks,” 2014. [Online]. Available: https://github.com/clemahieu/raiblocks
[11] D. J. Bernstein, N. Duif, T. Lange, P. Shwabe, and B.-Y. Yang, “High-speed high-security signatures,” 2011. [Online]. Available: http://ed25519.cr.yp.to/ed25519-20110926.pdf
[12] J.-P. Aumasson, S. Neves, Z. Wilcox-O’Hearn, and C. Winnerlein, “Blake2: Simpler, smaller, fast as md5,” 2012. [Online]. Available: https://blake2.net/blake2.pdf
[13] A. Biryukov, D. Dinu, and D. Khovratovich, “Argon2: The memory-hard function for password hashing and other applications,” 2015. [Online]. Available: https://password-hashing.net/argon2-specs.pdf