Daniele Piscaglia

L'algoritmo di consenso

L'algoritmo di consenso (consensus) è quel processo che in un sistema distribuito permette di raggiungere un accordo tra tutte le parti, senza che sia richiesto alcun rapporto di fiducia tra di esse. In una blockchain l'algoritmo di consenso deve garantire la sicurezza contro svariate problematiche, per cui diventa una scelta importantissima, ancora prima della implementazione stessa.

La parte più distintiva di un consesus è proprio il processo che viene effettuato per scegliere quale tra gli svariati nodi della rete, sarà colui che potrà validare[1] il prossimo blocco della catena. Questa scelta deve essere garantita tanto casuale quanto impossibile da controllare o manomettere, in maniera da evitare una futura convergenza della rete alla continua elezione di un gruppo ristretto di validatori che potrebbero quindi monopolizzare la catena. Uno degli aspetti che si rivelerà più complicati per il consensus sarà anche il poter garantire la convergenza di tutta la rete verso un'unica verità, cioè l'accordo di tutti i nodi nell'identificare una sola catena di blocchi, evitando quindi delle biforcazioni nel caso 2 validatori distinti vengano eletti nello stesso momento, portando quindi alla creazione di due blocchi in "competizione".

Proprietà che un buon algoritmo deve garantire:

- Sicurezza: tutti i nodi prodotti dai partecipanti (ed accettati) devono essere verificati validi dalle regole del protocollo
- Partecipazione: tutti i partecipanti dovrebbero partecipare in maniera attiva al costante processo di aggiornamento dello stato distribuito e in qualsiasi tipologia di votazione che venga effettuata
- Equalità: tra i vari voti ed il loro peso
- Costo: il costo o investimento necessario per acquisire il controllo della rete deve essere proporzionale al guadagno che si ottiene da esso, in maniera da renderlo economicamente sconveniente
- Verificabilità: Dovrebbe essere relativamente semplice per i partecipanti della rete poter verificare che tutto stia procedendo nella maniera legittima

Il non rispetto di queste direttive da parte di un algoritmo di consenso può potenzialmente esporre il sistema ad una serie di attacchi.

Attacco della doppia spesa

L'attacco di doppia spesa è il più importante che può essere utilizzato contro un ledger[2] distribuito. Prevede che l'attaccante effettui un pagamento ad un venditore per la ricezione di un bene o servizio. Una volta che il venditore può considerare confermato il pagamento e fornire il prodotto/servizio per cui ha ricevuto il pagamento all'attaccante, quest'ultimo provvede a

[1] Creare, autenticare e pubblicare
[2] Libro mastro, struttura dati

pubblicare una nuova transazione che annulla la precedente, in questo modo all'attaccante torneranno i fondi spesi che potranno essere riutilizzati in una nuova transazione.

Denial Of Service

Nel caso una organizzazione prendesse il controllo della validazione, quindi riuscisse a manipolare la lotteria dei validatori facendo in modo che il validatore eletto sia sempre uno sotto il suo controllo, allora potrebbe decidere di impedire l'aggiunta di un determinato tipo di transazioni oppure delle transazioni provenienti da un ente specifico, effettuando un operazione di censura.

Permissionless

Il campione di confronto

Con lo scopo di confrontare vantaggi e vulnerabilità, tra le soluzioni che sono già state proposte ed implementate, sono state prese in considerazione le 20 criptovalute attuali con la attuale capitalizzazione di mercato più alta. Decisione presa perché attualmente le criptovalute sono senz'altro l'implementazione più diffusa della blockchain, nonché le più delicate dal punto di vista di tutti i problemi che la decentralizzazione può portare riguardo la sicurezza e correttezza del sistema. Sono quindi state scelte quelle con capitalizzazione totale di mercato più alta dato che sono quelle che si possono considerare più sicure in quanto testate sul campo per un valore pari proprio alla capitalizzazione di mercato stesso e una qualsiasi falla porterebbe, con buona probabilità, al collasso della moneta stessa.
Prendendo quindi un campione delle prime 20 criptovalute per capitalizzazione di mercato si è trovata la seguente ripartizione nei relativi algoritmi di consenso:

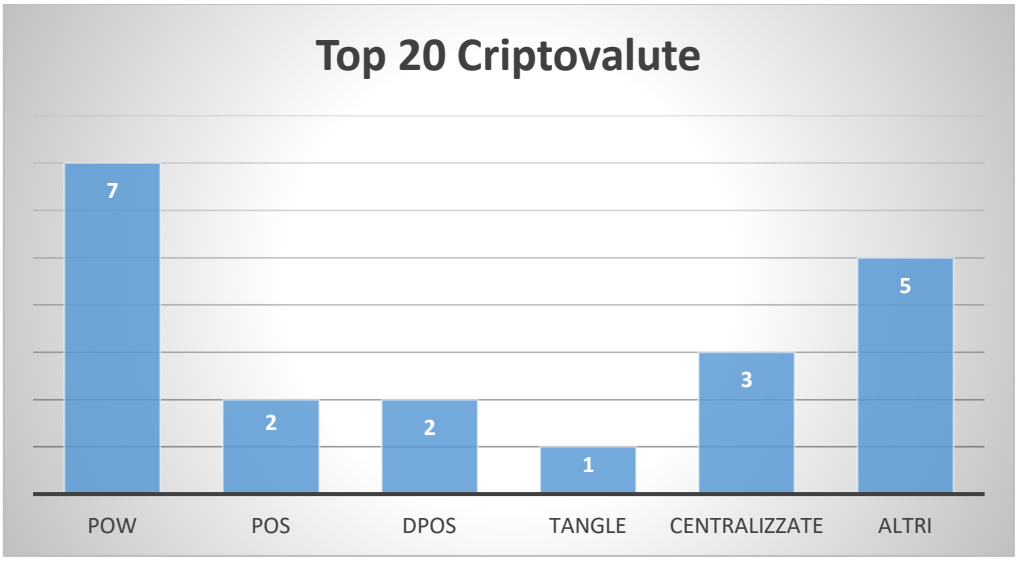

Figura 1 Consensus delle top 20 criptovalute (dati del 03/05/2019 da coinmarketcap.com)

Partendo proprio dalle statistiche raccolte sono stati scelti gli algoritmi che saranno posti a confronto (prendendo anche come esempio la implementazione più significativa per ogni algoritmo) focalizzandosi sulle problematiche di correttezza e sicurezza che ognuno introduce o risolve. In realtà solamente un sottoinsieme di consensus sono stati mantenuti in considerazione, sono state infatti ignorate tutte quelle implementazioni che sono considerate centralizzate, cioè che richiedono la presenza di un organismo centrale che ha dei poteri di controllo più o meno sovrani sulla blockchain e/o sulla scelta dei validatori (nello specifico Ripple, Stellar e Tether).

A questo punto la scelta si è focalizzata su 4 principali algoritmi:

- Proof of Work (PoW): il padre di tutti i consensus decentralizzati, è senz'altro il più testato sul campo grazie all'anzianità rispetto agli altri e soprattutto grazie al suo utilizzo nel famosissimo Bitcoin. Sarà considerato come termine di paragone principale dato che è attualmente visto come punto di riferimento da tutta la comunità
- Proof of Stake (PoS): è stata la prima alternativa proposta dopo il PoW, nonostante le potenziali problematiche che introduce sotto l'aspetto della sicurezza risulta una interessante alternativa soprattutto grazie al costo di funzionamento estremamente ridotto che comporta rispetto alla precedente
- Delegated Proof of Stake (DPoS): è in realtà una semplice variante del PoS che però merita una introspettiva tutta sua a causa delle differenze che comporta nella gestione della rete
- Tangle: è un approccio al problema completamente differente e rivoluzionario rispetto ai precedenti, che in realtà non è neanche considerabile come blockchain vera e propria ma che promette nuovi limiti di sostenibilità e scalabilità

Questo confronto permetterà di capire quanto sia realmente maturo e affidabile l'ambiente delle blockchain pubbliche (permissionless), quindi per fornire un quadro completo verranno anche analizzate le soluzioni più gettonate nell'ambito delle blockchain private (permissioned), tentando di evidenziare quali sono i vantaggi che un ambiente propone rispetto all'altro.

Proof of Work

Il primo algoritmo di consenso ad essere stato ideato ed implementato, che grazie alla sua semplicità garantisce uno standard di sicurezza molto elevato. Il processo di elezione del prossimo validatore si basa sulla risoluzione di un puzzle crittografico la cui soluzione può essere trovata esclusivamente per tentativi (in maniera simile ad un attacco in forza bruta). La probabilità di trovare la soluzione a questo puzzle (difficoltà) viene automaticamente gestita dalla rete in base alla potenza computazionale che ha a disposizione (che viene offerta dai partecipanti), questo permette di avere sempre un puzzle crittografico che risulti abbastanza complicato da richiedere appunto una quantità di tempo e di potenza di calcolo tale che per nessun attore sia possibile prendere il monopolio della rete. Basandosi questo meccanismo sulle regole matematiche che definiscono l'algoritmo di hashing che ogni implementazione usa e soprattutto sulla potenza di calcolo, che è quella risorsa fisica che introduce in un algoritmo software un costo ben reale determinato (in energie ed hardware), è possibile dare una prova concreta dell'affidabilità della rete.

In realtà però questa tipologia di rete non è priva di problemi, sia sotto l'aspetto della sicurezza che però come vedremo non costituiscono una reale minaccia, ma soprattutto per quanto riguarda la scalabilità e la sostenibilità della blockchain associata, che può rapidamente arrivare ad avere costi di mantenimento troppo alti per essere utilizzabile. Questa analisi non si concentra particolarmente su questo aspetto, ad ogni modo bisogna tenere presente che ad oggi non sia ancora stata trovata una reale soluzione per avere una blockchain versatile e scalabile che non comporti costi esagerati (si veda l'esempio della famosa criptovaluta ethereum che presto passera da un consenso PoW ad un PoS proprio per queste motivazioni).

51% Attack

È l'attacco più semplice in quanto intrinseco nel funzionamento dell'algoritmo stesso, consiste nel guadagnare una quantità di potenza computazionale superiore al resto dell'intera rete consentendo quindi all'attaccante di creare nuovi blocchi più velocemente di quanto possano gli altri nodi. In questo modo diventa per lui possibile in qualsiasi momento creare una biforcazione della catena (mantenendola inizialmente in locale) che però avendo più potenza computazionale rispetto al resto della rete riuscirebbe a produrre blocchi più velocemente di quella "legittima" e quindi non appena pubblicata sarebbe accettata da tutti i nodi come nuova catena (in quanto più lunga). Questo scenario rende facili degli attacchi di doppia spesa semplicemente rimuovendo dalla catena da lui creata quei pagamenti che vuole eliminare. Nella stessa maniera è possibile per l'attaccante anche censurare e gestire come meglio crede i pagamenti degli altri partecipanti, eliminando tutti quelli che non vuole accettare.

Questo scenario ad ogni modo diventa sempre più improbabile con il crescere della popolarità di una moneta e di conseguenza dell'interesse dei partecipanti della rete (con relativa potenza computazionale), portando il costo reale di un ipotetico attacco del 51% ad un valore tale da non renderlo più appetibile.
Sul sito crypto51 [1] sono reperibili dati aggiornati sul costo che un eventuale attacco del 51% richiederebbe su alcune blockchain.

Proof of Stake

Il consensus proof of stake è senz'altro il più diffuso dopo il proof of work, dal quale si distingue principalmente per aver escluso completamente l'utilizzo di una risorsa esterna per la creazione di nuovi blocchi. Mentre il PoW richiede un costante consumo di potenza computazionale, il PoS si basa sulla quantità di monete depositate da ogni validatore nello "stake", scegliendo in maniera casuale i prossimi validatori.

Questo comporta degli importanti vantaggi:

- Per ogni account non si ha nessun vantaggio nel tentare la produzione di un blocco per più di una volta al secondo, quindi la rete ha un consumo paragonabile alle altre tradizionali infrastrutture distribuite (come ad es. BitTorrent)
- Permette uno standard di sicurezza più alto rispetto al PoW, infatti considerando il costo computazionale necessario per un 51% attack nel PoW paragonabile alla somma dei 'reward' e delle commissioni dei blocchi che vengono prodotti nella finestra di tempo dell'attacco, la controparte in PoS sarebbe un costo molto più alto, pari al 51% del valore totale di tutta la liquidità della moneta.
- In base alla specifica implementazione può arrivare a permettere un numero di transazioni al secondo molto più alto (più blocchi al secondo, contro il blocco ogni minuto di Ethereum o addirittura ogni 10 minuti di bitcoin)

Ad ogni modo però, nonostante i validatori siano incentivati all' "onestà" per non perdere le loro monete nel deposito, nascono numerose potenziali problematiche, alcune di carattere più economico (come la distribuzione iniziale delle monete, il maggiore controllo della rete da parte dei più "ricchi" ecc.) ed altre che minano alla sicurezza ed affidabilità della rete. Di conseguenza saranno necessari degli accorgimenti particolari rispetto alle politiche che sono state affrontate nel caso del PoW.

Sidechaining

Avendo reso la produzione di blocchi a tutti gli effetti gratuita (non è più necessario risolvere un complicato puzzle criptografico) è ora teoricamente possibile per un qualsiasi stakeholder[3], partendo dal blocco attuale o da un qualsiasi punto precedente, minare in successione una nuova catena di blocchi, appositamente manipolata per rieleggere ripetutamente dei validatori favorevoli. Questo procedimento ovviamente richiederebbe, in maniera simile al PoW la risoluzione di un vero e proprio puzzle criptografico che però in questo caso sarebbe completamente centralizzato e non in competizione con nessuno.

Nothing at stake

Questo problema si propone ad ogni possibile biforcazione della rete. In un PoW ogni miner deve scegliere su quale catena direzionare la propria potenza computazionale (e quindi "scommettere" su di essa), nel PoS invece nessuno gli vieta di puntare su entrambe le catene (nessun costo in risorse reali, al contrario del costo reale che la potenza computazionale ha nel PoW), qualsiasi si

[3] Partecipante attivo della rete

riveli vincitrice a lui non interessa, ed è quindi incentivato a scommettere su qualsiasi catena lui veda, potenzialmente portando ad una situazione dove la rete non raggiunge mai un consenso su quale sia la catena "giusta".

Questo scenario potrebbe inoltre rendere una doppia spesa possibile possedendo un piccola quantità di liquidità (anche fino all'1%) nell'eventualità di una qualsiasi biforcazione della catena. Supponendo infatti che tutta la rete continui a "supportare" entrambe le biforcazioni, un utente malevolo A potrebbe inviare un pagamento ad un utente B nella prima catena, contemporaneamente allo stesso pagamento a se stesso nella seconda catena. Non appena il pagamento venisse confermato dall'utente B, quindi ridirezionare tutta la sua puntata sulla catena dove il pagamento era in realtà "annullato", fornendo a questa catena quella piccola quantità di supporto in più che eventualmente la porterà in vantaggio sull'altra facendola quindi scomparire con il pagamento che era stato inviato all'utente B [2].

Per quanto questo problema possa sembrare piuttosto impossibile nella pratica, o quanto meno richiedere una influenza sulla rete ben superiore all'1% ipotizzato, è stato preso piuttosto sul serio dalla comunità degli sviluppatori, che ha portato all'individuazione di vari approcci al problema:

- Introdurre nel protocollo un modo per penalizzare chiunque effettua un multiplo voto su due o più catene
- Introdurre nel protocollo un modo per penalizzare chiunque effettua un voto su una catena che poi si rivela essere "errata". Meccanismo utilizzato da Ethereum (dopo l'aggiornamento CASPER) dove un voto multiplo porterebbe alla perdita di tutto il deposito (che ha durata 4 mesi) di un validatore.
- Decidere in anticipo quali sono i nodi che potranno validare i blocchi di un fork, in questo modo non sarà possibile per un validatore malevolo condizionare la scelta di una catena piuttosto che un altro spostandosi da una parte piuttosto che da un'altra.

Long Range Attack

Nella tipologia degli attacchi long range (a lungo raggio) ricadono tutti quegli ambiziosi tentativi di riscrivere l'intera blockchain a partire dal genesis block[4] o da un blocco non recente della blockchain [3]. In un PoW un attacco del genere non avrebbe senso in quanto non solo richiederebbe all'attaccante di sopraffare la potenza totale attuale della rete, bensì di compensare anche precedente fino a partire dal primo blocco che andrebbe ricalcolato. Lo scenario in un caso Proof of Stake invece è ben diverso, in quanto per l'attaccante la creazione di un'altra catena (partendo anche dal genesis block) avrebbe un costo quasi nullo. Ora supponendo che l'attaccante sia in controllo di una minima parte del deposito totale, di conseguenza sulla sua catena "fraudolenta" (dove è l'unico partecipante) è evidente che solo una minima parte dei blocchi verrebbero effettivamente prodotti (dato che tutti gli altri validatori non partecipano alla rete) portando quindi la catena "fraudolenta" ad essere decisamente più corta di quella "legittima" rendendola facilmente riconoscibile. Ad ogni modo in questa catena risultando l'attaccante come unico validatore con l'avanzare del tempo guadagnerebbe in maniera esponenziale su tutti gli altri un vantaggio nella quantità del deposito (grazie alle penalizzazioni che gli altri validatori

[4] Primo blocco della blockchain. Nel caso del Proof of Stake risulta particolarmente importante in quanto è quello che contiene i dati dei depositi dei validatori iniziali.

riceverebbero in quanto non partecipanti alla rete e alle commissioni delle transazioni che potrebbe simulare o ricopiare dalla catena principale) fino a permettergli di ottenere il monopolio della sua rete. A questo punto per l'attaccante diventa semplice (essendo l'unico validatore) espandere la propria blockchain fino a che questa risulti più popolata e quindi venga scelta al posto di quella "legittima". Nulla vieta all'attaccante di inserire tutte le transazioni normalmente prelevate dalla catena principale e ad esempio di ripristinare i valori di tutti i depositi ai valori originali rendendo così la nuova blockchain "accettabile" anche da un punto di vista delle ripartizioni dei fondi in modo da rendere l'attacco molto meno evidente.

Per evitare la possibilità di trasportare le transazioni da una blockchain all'altra è possibile inserire **all'interno della transazione stessa un riferimento al blocco precedente** (della blockchain principale) in maniera che non possa essere trasportata sulla blockchain fraudolenta (dove il blocco non è presente) rendendo un eventuale attacco quantomeno evidente.

Lo scenario appena descritto diventa ancora più preoccupante considerando che in realtà partendo dal genesis block gli stakeholder in esso contenuti potrebbero in realtà aver già speso il proprio deposito, rendendoli quindi facilmente corrompibili (**Posterior Corruption**) da un qualsiasi attaccante, che se entrasse in possesso delle loro chiavi avrebbe ancora più possibilità di successo nell' attacco. Una possibile contromisura potrebbe essere quella di utilizzare una Key Evolving Signature[5] ad ogni modo non ci sono attualmente meccanismi per verificare che un utente elimini effettivamente la chiave privata una volta che questa venga utilizzata, impedendogli di "rivenderla" al miglior offerente in futuro.

Al momento l'unica soluzione efficace contro questa tipologia di attacchi sembra quindi quella di utilizzare alcuni blocchi di **checkpoint** cioè oltre i quali non è possibile tornare indietro. Questo permetterebbe quindi le biforcazioni della catena a ritroso solamente fino ad un numero prefissato *k* di blocchi. Nel caso in qualsiasi momento una nuova catena apparisse con un punto di origine precedente agli ultimi *k* blocchi questa verrebbe automaticamente ignorata. Questo meccanismo non pone nessun problema per un nodo che è sempre online sulla rete, ma risulta un problema per quei nodi che si connettono per la prima volta o vanno offline per una finestra di temp. Questo porta ad una **Weak Subjectivity** di quei nodi che alla riconnessione dovranno andare a recuperare i blocchi "persi" e con essi anche gli eventuali checkpoint, dovendosi quindi "fidare" di chi glieli fornisce [4].

In conclusione un sistema PoS può quindi proteggersi solamente da uno tra i problemi di Posterior Corruption oppure Weak Subjectivity in quanto la protezione da uno comporta l'esposizione all'altro.

[5] Ogni volta che una chiave privata viene utilizzata per firmare qualcosa questa viene eliminata e quindi sostituita da una nuova chiave per la prossima firma.

Ethereum

Attualmente rete principale ancora in PoW ma in atto un primo affiancamento e successiva trasformazione in un meccanismo PoS. Richiederà ai validatori un deposito che sarà bloccato per una finestra di 4 mesi, per risolvere il problema della corruttibilità dei validatori dopo questo periodo di tempo utilizzerà dei checkpoint periodici.

Cardano

Per evitare che la maggior parte dei partecipanti, soprattutto quelli con un piccolo patrimonio, non partecipino attivamente alla rete è permesso a quest'ultimi di "delegare" le proprie monete a un altro nodo che quindi acquisirà una potenza di voto proporzionale alle monete delegate. Questa soluzione non va confusa con il Delegated Proof of Stake che verrà visto di seguito, dove il processo di elezione è completamente differente.

Per evitare che un validatore diventi facilmente corrompibile appena gli è permesso di ritirare il proprio deposito utilizza una Key Evolving Criptography che ad ogni utilizzo della chiave propria chiave privata per la validazione di un blocco rende quest'ultima non più valida e ne crea una nuova da utilizzare al suo posto [5]. Non è quindi richiesta la sincronizzazione di un eventuale nodo che torna online dopo un lungo periodo offline tramite l'utilizzo di checkpoint (risolvendo la Weak Subjectivity) ad ogni modo non esiste alcun meccanismo per assicurarsi che un validatore provveda effettivamente ad eliminare una chiave privata dopo il suo utilizzo.

Delegated Proof of Stake

Il meccanismo Delegated Proof of Stake, pur sempre basandosi sulla quantità di monete possedute (come il PoS) presenta uno schema di voto e di gestione completamente diverso, focalizzato sull'elezione di alcuni elementi che gestiranno la validazione dei nuovi blocchi. Un altro aspetto molto importante di questo meccanismo è il fatto che questi vincitori delle elezioni saranno anche coloro che in caso di proposte per la modifica della rete potranno accettarle o rifiutarle.

Ogni stakeholder della rete riceve una quantità di voti proporzionali al suo capitale, che può quindi utilizzare per votare uno o più validatori. I validatori che ricevono più voti vengono eletti come testimoni (da 21 fino a 101 in base alle implementazioni) a questo punto sono solamente i testimoni che a turno producono i nuovi blocchi, che essendo in un numero relativamente molto piccolo possono svolgere il compito molto più velocemente (rispetto a PoW e PoS). I testimoni vengono ripagati tramite commissioni che, nella maggior parte dei casi, vengono poi condivise anche con i rispettivi elettori. È richiesto che i nodi intenti a diventare testimoni siano sempre online e disponibili, infatti nel caso un testimone risulti non disponibile (non produca un blocco che spettava a lui) o ancora peggio si rivelasse disonesto è molto semplice per tutti i votati spostare il proprio sostegno su qualcun altro e sostituirlo, scenario che quindi incentiva tutti i testimoni all'onestà (in base alle implementazioni potrebbe anche essere richiesto ai testimoni di effettuare un deposito che verrebbe bloccato per una quantità di tempo).

La potenza di voto per l'elezione è sempre calcolata sulla base della moneta (stake) posseduta da ogni partecipante, ad ogni modo affidando la produzione dei nuovi blocchi ai soli testimoni vengono risolti tutti i precedenti problemi visti nel caso del PoS. Le proprietà che un nodo deve soddisfare per essere eletto però possono spesso essere molto costose e stringenti permettendo solo a pochi di soddisfarle e di conseguenza portare spesso alla rielezione degli stessi elementi. Sotto un certo punto di vista può essere considerata la soluzione più democratica in quanto tutti i partecipanti hanno modo di esprimere il proprio voto e in maniera più o meno diretta partecipare quindi al processo di consenso, ad ogni modo è anche evidente che una volta che i testimoni sono stati scelti possiedono il completo monopolio della rete (almeno fino alla prossima rielezione).

Proprio per questo motivo, solitamente i testimoni per poter essere votati e considerati "affidabili" devono identificarsi (come persona, azienda o autorità), anche per impedire un eventuale "sybil attack", dove una singola autorità potrebbe partecipare alla rete sotto più nomi, facendo apparire come competitori vari testimoni che in realtà apparterrebbero allo stesso proprietario. E' quindi evidente che una blockchain DPoS risulta essere molto simile ad una blockchain Proof Of Authority come può essere Ripple dove però anziché esserci una singola autorità centrale ce ne sarebbero un numero prefissato, elette (e modificate) periodicamente. [6]

EOS

Nonostante la piattaforma sia una tra le ultime arrivate occupa il quinto posto in classifica (di valore) ed ha attirato molte attenzioni sin da prima del suo rilascio (è la criptomoneta ad aver avuto la ICO[6] più grande), in quanto grazie alla sua implementazione DPoS a 21 testimoni si

[6] Initial Coin Offering: vendità della moneta prima che la rete venga rilasciata per garantire una spartizione equa del capitale

presenta come una delle blockchain più scalabili ed economiche per ospitare delle applicazioni decentralizzate [7].

La disputa che più affligge questa blockchain è il fatto che in realtà non sembra essere decentralizzata come promette, avendo dimostrato più volte di poter effettivamente cancellare o modificare delle azioni che erano state applicate e verificate sulla blockchain. Questi "rollback" sono stati effettuati tramite l'intervento di "arbitri" che si assumevano la responsabilità di analizzare dei casi di errore (bug nella rete) o fraudolenti (come il furto) che facendo rapporto ai testimoni potevano modificare lo stato come ritenevano più giusto. Un altro esempio è la possibilità della rete di poter confiscare e ridistribuire i fondi di un partecipante dopo un determinato periodo di inattività [8]. La possibilità di modificare lo stato della blockchain (che per sua natura dovrebbe essere un'entità immutabile), nonostante fino ad ora sia stato fatto solo per una "giusta" causa, ha ovviamente sollevato non poche discussioni sul quanto possa essere utile una blockchain (che sono nate per permettere la sovranità individuale tramite la decentralizzazione) se in realtà può essere controllata e modificata.

DAG

Forma del dag, tips e pesi (pow su ogni nuovo site)

I DAG (grafi aciclici diretti) presentano un nuovo approccio completamente differente da quello delle blockchain. Nei DAG infatti non è presente una singola catena di blocchi ma esistono numerose ramificazioni di "blocchi". In questo caso specifico consideriamo una implementazione particolare di DAG, cioè quella scelta da IOTA [9].

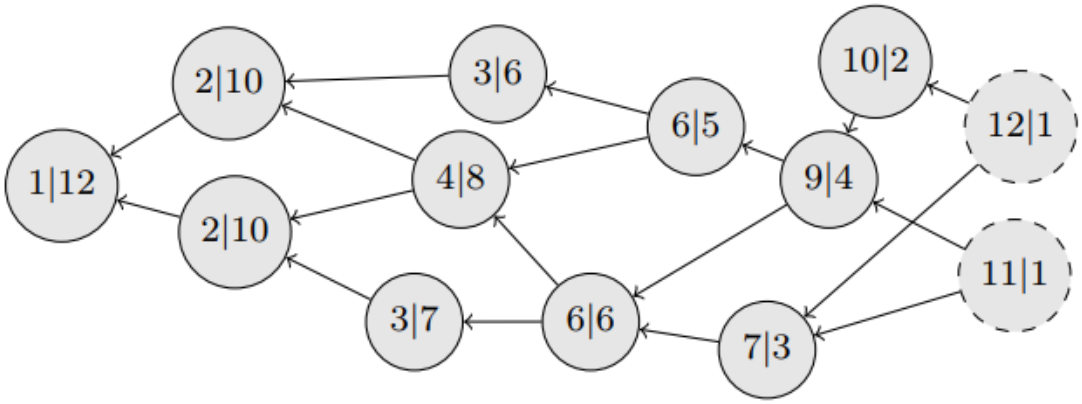

Figura 2 Esempio di DAG dove i numeri a destra in ogni sito indicano il peso cumulativo di ogni nodo. I due siti con i bordi tratteggiati sono le transazioni ancora non confermate (tips)

Chiamiamo blocco o sito ogni elemento del DAG che rappresenta una transazione, ogni transazione per essere accettata (aggiunta) deve verificare e confermare due transazioni già presenti nel DAG chiamate *tips*. Queste *tips* sono casualmente scelte tra tutte le transazioni che non risultano ancora confermate, cioè che non hanno ancora almeno 2 transazioni figlie che le referenziano. Per ogni transazione è associata una quantità di lavoro computazionale (PoW), nella figura sopra tutte transazioni hanno un peso pari a 1. Per ogni transazione si ha quindi un peso cumulativo che è pari al proprio peso, più quello di tutte le transazioni che la hanno confermata, quindi il peso cumulativo equivale al quantitativo di PoW che è stato svolto per confermare quella transazione.

La scelta delle tips da parte di tutti i nodi della rete deve avvenire in maniera casuale ed utilizzando lo stesso metodo decisionale, chiamato TSA (Tip Selection Algorithm). Questa scelta svolge una parte importante nel sistema in quanto è quella che decide quale parte del DAG dovrà espandersi; IOTA utilizza una scelta chiamata Monte Carlo Markov Chain (MCMC) che permette di scegliere le tips in base al peso cumulativo del ramo che occupano, in questo modo i rami che tenderanno ad avere più transazioni e più PoW associato saranno quelli che si espanderanno sempre di più mentre gli altri rami tenderanno ad essere abbandonati.

Il vantaggio principale di questo nuovo approccio è il fatto che le transazioni stesse vengono confermate dalle nuove transazioni in arrivo (ogni nuova transazione ne conferma 2 precedenti), rendendo quindi la velocità del sistema sempre maggiore al crescere del numero dei partecipanti. Inoltre essendo le transazioni stesse delle validatrici non c'è bisogno di alcun miner (mantenitore/validatore della rete) e ciò permette alle transazioni di essere completamente gratuite (senza commissioni), dove quindi il totale importo inviato è pari a quello ricevuto, rendendo il sistema perfetto per l'utilizzo in un ecosistema a micropagamenti.

Assiduous Honest Majority Assumption

Come si è visto in precedenza questo sistema richiede l'utilizzo del PoW solamente quando si intende aggiungere una nuova transazione al grafo, questo evita gli esorbitanti sprechi di potenza computazionale che sono richiesti negli ortodossi sistemi PoW. Questo però significa anche che in ogni momento la potenza di calcolo totale associata alla sicurezza della rete non è quella di tutti i partecipanti, bensi quella che viene utilizzata per l'inserimento delle nuove transazioni. Come già detto, grazie al TSA utilizzato da IOTA (MCMC), i rami che tendono a svilupparsi più in fretta sono quelli che hanno associata una maggiore quantità di lavoro PoW svolto, quindi un eventuale attaccante dovrebbe riuscire a sopraffare, con l'utilizzo della propria potenza computazionale, solamente quella parte di nodi "onesti" che sono attualmente attivi sul sistema.

Questo problema però viene risolto nel tangle di IOTA dall'algoritmo MCMC per la scelta delle tips, infatti per un attaccante sarebbe impossibile direzionare la proprio potenza di calcolo solo sul ramo che alimentare, in quanto il MCMC lo forzerebbe a scegliere le tips anche dal ramo onesto, quindi la sua potenza andrebbe inevitabilmente anche ad alimentare la parte onesta della rete [10].

Permissioned

In questa sezione verranno analizzate le soluzioni di blockchain private più famose. La principale differenza negli algoritmi di consenso tra il mondo pubblico e privato è che mentre il primo si basava su un concetto monetario (in termini di distribuzione nel PoW come ricompensa per i miner, o in termini di *potenza di voto* nei sistemi PoS) nell'ambito privato, spesso ad una blockchain non è associata nessuna moneta. Solitamente i partecipanti di queste reti sono economicamente incentivati alla partecipazione dall'ecosistema stesso che la determinata blockchain gestisce e non da un eventuale guadagno diretto che essa comporta, ad esempio monetario come nel caso delle blockchain pubbliche. Questo ovviamente richiede degli algoritmi di consenso completamente diversi che non possono più fare affidamento sull'onesta dei nodi in base ad un vantaggio economico come succedeva nelle blockchain pubbliche.

In questo tipo di blockchain quindi è richiesta una autenticazione da parte dei nodi validatori (e solitamente anche dei clienti che la utilizzano), garantendo così la conoscenza del numero di nodi che partecipano alla validazione e della loro identità/appartenenza. Questo dettaglio non implica o richiede alcun fattore di fiducia tra le organizzazioni coinvolte, ma comunque, grazie all'identificazione permette di fare delle assunzioni e di evitare molti problemi che andrebbero affrontati in maniera differente nel caso delle blockchain permissionless (come il sybil attack).

Non avendo quindi una capitalizzazione di mercato sulla quale basarsi per la classificazione degli algoritmi più importanti (come è stato fatto in precedenza), l'unica metrica possibile per la scelta è stata quella del grado di diffusione e interesse da parte della comunità.

Practical Byzantine Fault Tolerance

Questo meccanismo di consenso prevede una sincronizzazione diretta tra tutti i nodi che partecipano alla validazione, permettendo di raggiungere una validazione immediata appena una transazione viene aggiunta alla blockchain (senza dover attendere la conferma da parte di più blocchi consecutivi). In questo scenario, ogni client invia la propria transazione ad un qualsiasi nodo della rete che diventa quindi il creatore della transazione.

A questo punto ogni transazione viene divisa in quattro passi:

- Pre-prepare: inviato dal creatore della transazione a tutti gli altri nodi (anche quelli offline)
- Prepare: Ogni nodo che riceve un messaggio di pre-prepare (che sia verificato valido), inoltra a tutti gli altri nodi un messaggio di prepare
- Commit: ogni nodo procede all'effettiva esecuzione della transazione (solo nel caso abbia ricevuto almeno 2*f prepare/pre-prepare messaggi, dove f è il numero di nodi "errati") ed invia la commit (conferma) per comunicarlo a tutti gli altri nodi
- Reply: ogni nodo che ha confermato la transazione invia una reply al client per informarlo, nel caso il client riceva almeno f+1 risposte è sicuro che la transazione è stata validata

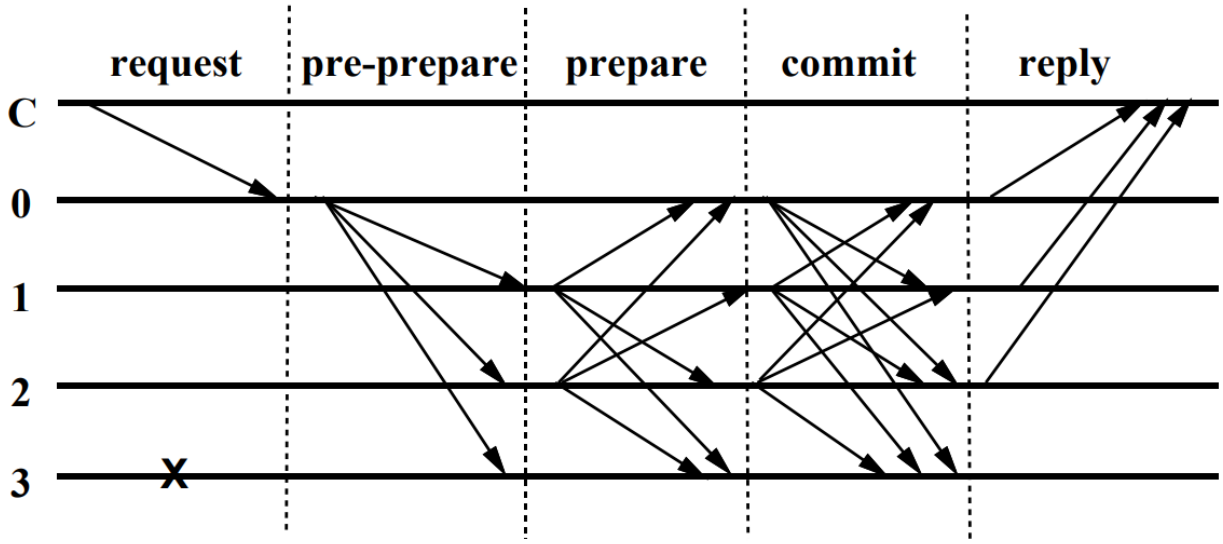

Figura 3 Esempio di interazione pBFT con 4 nodi dove il nodo 3 va in errore durante la prima fase [11]

Nel caso rappresentato in figura quindi (con n=4 nodi validatori ed f=1 nodo byzantine fault), per la gestione della transazione sono generati:

- **1** messaggio di **request** dal client al nodo validatore che riceve la richiesta
- 3 **(n-1)** messaggi di **pre-prepare** dal nodo creatore a tutti gli altri
- 6 **(n-1-f)*(n-1)** messaggi di **prepare** tra i vari nodi attivi (non in crash)
- 9 **(n-f)*n** messaggi di **commit** tra i nodi per la conferma della transazione
- 3 **(n-f)** messaggi di **reply** per comunicare al client l'effettivo avvenimento della transazione

Questo ammonta ad un totale di 22 messaggi (33 nel caso non ci fossero stati nodi in crash) già con soli 4 nodi validatori.

Risulta ovvia la scarsa scalabilità al crescere di n (numero di nodi partecipanti) dato che il protocollo richiede che tutti i nodi comunichino direttamente tra di loro, ma con un numero

limitato è molto efficiente (soluzione BFT-NFS circa il 3% più lenta di una soluzione centralizzata [12]) e non richiede alcun tempo di conferma dopo che una transazione è stata validata da tutti i nodi.

Il problema del sybil attack, simile a quello già visto nel caso DPoS, si riferisce al controllo o coalizione di un numero f di nodi validatori, infatti il protocollo di questa rete risulta sicuro fino ad un numero massimo di nodi compressi f su un totale di *3*f+1* nodi. Questo aspetto diventa sempre più difficile da violare al crescere del numero di nodi partecipanti (similmente a quanto succede nelle altre blockchain), data la limitata scalabilità del protocollo però solitamente questo risulta essere un problema.
Per questo motivo solitamente si utilizza una strategia diversa, come può essere quella di affidarsi alle autorità dei nodi, impedendo l'appartenenza di più nodi alla stessa figura e un insieme di organizzazioni che tramite un qualche legame di competitività al di fuori della blockchain impediscano la possibilità di una collaborazione e manipolazione della blockchain che porterebbe a loro svantaggio. In altri casi invece viene affiancato un altro meccanismo di consenso come può essere quello di effettuare delle fasi PoW ogni numero prefissato di blocchi.

Proof of Elapsed Time

Questo algoritmo può essere visto come una sorta di Proof Of Work con un funzionamento differente della lotteria, dove ogni nodo, anziché effettuare costosi calcoli computazionali, viene messo in attesa, permettendogli quindi di svolgere altri calcoli per altre applicazioni. Questo rende l'algoritmo molto efficiente e poco costoso, risolvendo tutti i problemi di sostenibilità legati al POW.

A questo scopo, per l'elezione del nuovo validatore di ogni blocco ognuno dei nodi esegue l'estrazione di un tempo di attesa casuale t, a questo punto si mette in attesa per t tempo. Il primo validatore a risvegliarsi potrà pubblicare il nuovo blocco.

Il corretto funzionamento della rete è ovviamente condizionato dalla corretta esecuzione dei passi dell'algoritmo da parte di tutti i partecipanti. Nello specifico è necessario potersi assicurare che ogni partecipante:

- Estragga veramente in maniera casuale il tempo di attesa, senza condizionare l'estrazione in modo da estrarre un tempo corto per diventare il vincitore
- Attenda realmente per la quantità di tempo che viene estratta

Questi aspetti di sicurezza e verificabilità della rete sono risolti dall'utilizzo di un'enclave Intel SGX ®. Questo algoritmo di consenso infatti sfrutta le proprietà di un enclave:

- Sicurezza: Il codice in essa eseguito non può essere modificato
- Verificabilità: Per il codice eseguito viene generato un certificato che ne dimostra la corretta esecuzione

Questa soluzione è stata implementata nella blockchain Permissioned Hyperledger Sawtooth.

Proof of Majority

Algoritmo che viene presentato [13] come una nuova soluzione molto efficiente, dove i nodi anziché competere in una elezione per stabilire il nuovo generatore di un blocco, collaborano contemporaneamente producendo lo stesso blocco. La produzione concorrente dello stesso blocco presso più nodi, formandone più copie è il processo stesso di conferma, che si considera completato quando il numero di copie è tale da raggiungere la maggioranza nella rete.

A questo scopo, per ogni blocco vengono definite le seguenti proprietà (incluse nell'header):

- Numero di transazioni contenute nel blocco
- Peso (*weight*) del blocco
- Conferma (*finalized*) del blocco

Ogni nodo partecipante al processo, riceve tutte le transazioni che devono essere aggiunte alla blockchain e quindi procede alla creazione del blocco stesso in maniera completamente autonoma. Ogni nuovo blocco viene creato con il valore di peso (*w*) a 1 e la conferma (*f*) a false, che indica che il blocco non è ancora da considerarsi "valido". Questo processo viene ripetuto da tutti i nodi sulla rete, portando quindi (esclusi eventuali problemi di partizionamento) alla creazione di numerosi blocchi identici. Successivamente, tutti i blocchi con lo stesso hash (gemelli) possono quindi essere uniti (merge) in un unico blocco che però avrà un peso (*w*) pari al numero di blocchi identici che lo "confermano" e potrà quindi essere finalizzato (campo *finalized* con valore true).

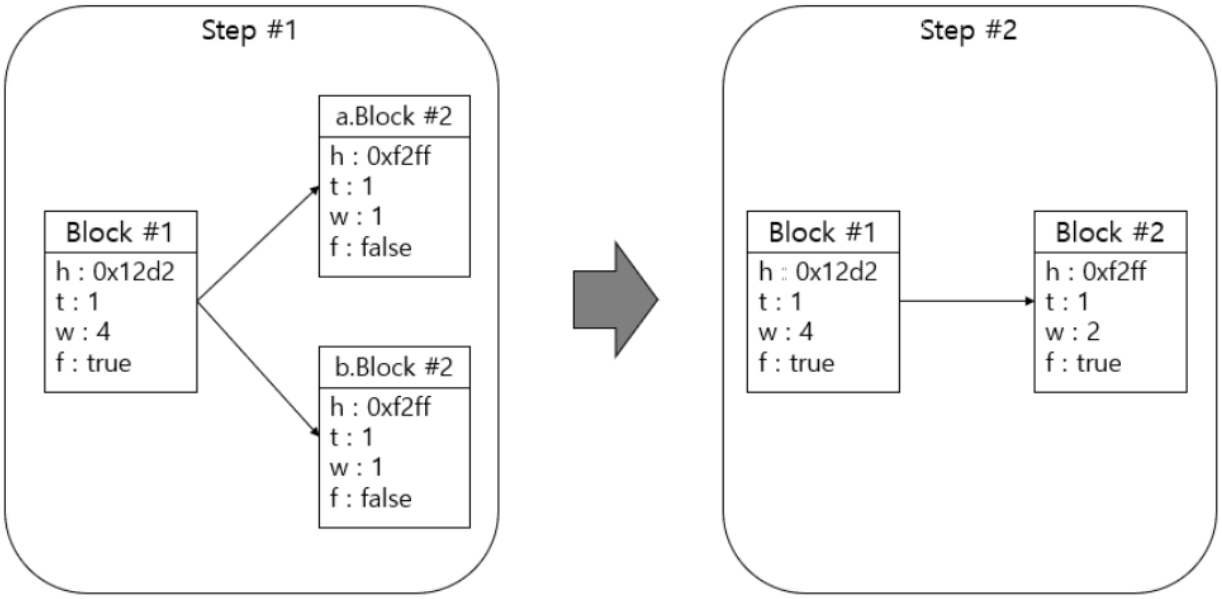

Figura 4 Processo di creazione e conferma di un blocco

È tutta via possibile che per qualche motivo, nella rete si verifichi un fork[7] nel quale vengono creati contemporaneamente due blocchi contrastanti (vedi figura sotto), in tal caso quindi, dopo il merge tra le varie versioni in competizione verrà selezionata quella con il peso più alto e solamente quella verrà confermata impostando il valore finalized a true mentre l'altra verrà ignorata.

[7] Biforcazione, contrasto di due "versioni" contrastanti sulla blockchain, cioè due blocchi diversi che dovrebbero occupare la stessa posizione

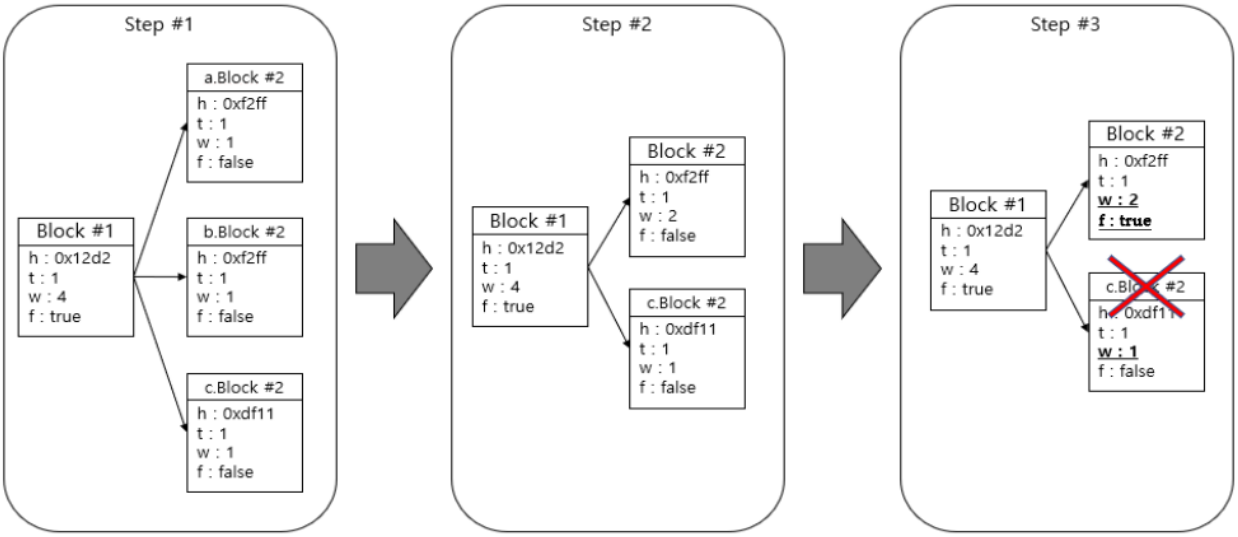

Figura 5 Processo di convergenza dopo un fork

Questo meccanismo di consenso, rimanendo molto semplice si avvale di tutti i vantaggi che una blockchain privata fornisce, senza introdurre la necessità di sfruttare calcoli complessi per garantire la sicurezza, sfruttando la conoscenza delle identità dei nodi per evitare attacchi di tipo Sibyl visti in precedenza e permette di garantire la sicurezza fino a quando il numero di nodi onesti supera il numero di nodi disonesti. Inoltre non è necessario alcuno scambio di messaggi per la sincronizzazione tra i nodi (se non il semplice trasferimento di nuove transazioni e nuovi blocchi) rendendo il sistema facilmente scalabile e performante.

Riferimenti

[1] «Crypto51.app,» [Online]. Available: https://www.crypto51.app/.

[2] J. Martinez, «Understanding Proof of Stake: The Nothing at Stake Theory,» [Online]. Available: https://medium.com/coinmonks/understanding-proof-of-stake-the-nothing-at-stake-theory-1f0d71bc027.

[3] A. Sharma, «Understanding Proof of Stake through it's Flaws,» [Online]. Available: https://medium.com/@abhisharm/understanding-proof-of-stake-through-its-flaws-part-3-long-range-attacks-672a3d413501.

[4] E. Deirmentzoglou, «Rewriting History: A Brief Introduction to Long Range Attacks,» [Online]. Available: https://blog.positive.com/rewriting-history-a-brief-introduction-to-long-range-attacks-54e473acdba9.

[5] A. Kiayias, «Ouroboros Genesis: A Provably Secure Proof-of-Stake Blockchain Protocol,» [Online]. Available: https://www.youtube.com/watch?v=LCeK_4o-NCc.

[6] [Online]. Available: https://hackernoon.com/why-dpos-is-not-really-a-pos-but-rather-a-poa-protocol-5bb1aa305625.

[7] B. Xu, «EOS Test Report,» [Online]. Available: https://www.whiteblock.io/library/eos-test-report.pdf.

[8] «EOS Termination agreement,» [Online]. Available: https://i.redd.it/tunpro4218411.png.

[9] «IOTA Tangle paper,» [Online]. Available: https://assets.ctfassets.net/r1dr6vzfxhev/2t4uxvsIqk0EUau6g2sw0g/45eae33637ca92f85dd9f4a3a218e1ec/iota1_4_3.pdf.

[10] «Stability in the tangle,» [Online]. Available: https://hal.archives-ouvertes.fr/hal-01716111v2/document.

[11] «pBFT paper,» [Online]. Available: http://pmg.csail.mit.edu/papers/osdi99.pdf.

[12] M. Castro e B. Liskov, «BFT can be fast,» [Online]. Available: https://www.researchgate.net/publication/3908996_Byzantine_fault_tolerance_can_be_fast/link/575a8f9e08ae9a9c955165fe/download.

[13] J.-T. Kim, J. Jin e K. Kim, «A study on an energy-effective and secure consensus algorithm for private blockchain systems,» [Online]. Available: https://ieeexplore-ieee-org.ezproxy.unibo.it/document/8539561.