Bitcoin

- Protokol pro distribuovaný konsenzus.
- Vytvořen v 2009 skupinou Sastoshi Nakamoto.
- Používá proof of work.
- Search puzzle
  - Mějme hashovací funkcí $H : \{0,1\}^* \to \{0,1\}^n$
  - Puzzle ID $\leftarrow$ D, kde D má dostatečnou min. entropii.
  - Cílová množina $S \subset \{0,1\}^n$ Řešením je x t.ž $H(D||x) \in S$.
  - Podle velikosti S můžeme určit obtížnost. Pro $S = \{0,1\}^n$ je řešení triv. Pro $S =$ jeden řetězec nemusí řešení existovat.
  - Pro nalezení řešení počet evaluací H odpovídá velikosti S.
  - Pro ověření řešení stačí jen jedno vyhodnocení funkce H.

Blockchain

- Spojový seznam s hash pointery na předchozí prvek v seznamu.
  - Jeho výhodou je, že omezuje ex-post manipulaci.
    Čím hlouběji bychom chtěli editovat seznam, tím víc kolizí H musíme řešit.
- Hešovací funkce SHA-256
- Elektronický podpis
  - Alg. Ecliptic Curve Digital Signature
  - Místo Z_n^* používá množinu odpovídající eliptické křivce.
- Registrace uživatele
  - Nový uživatel vygeneruje soukromý a veřejný klíč pro SHA-256, jeho identita odpovídá veřejnému klíči.
  - Jeden uživatel může mít více identit.
- Konsenzus
  - Je třeba se dohodnout na platné historii transakcí.
  - Problém byzantských generálů
    - Generálové na různých místech se mají dohodnout na tom, zda zaútočit nebo se stáhnout. Zprávy o jejich rozhodnutí se můžou ztratit a navíc mezi sebou můžou mít zrádce, který může tvrdit různým generálům různé věci.
    - Dokázalo se, že pokud je zrádců méně než polovina, tak lze tento problém vyřešit.
    - Řešení nelze použít pro BTC, protože předpokládají, že nikdo nemůže poslat zprávu jako někdo jiný a navíc si nemůže vytvořit více identit a přehlasovat tak ostatní.
  - Permissionless setting - kdokoliv se může zúčastnit transakcí a ověřování bloků.

Hlavní myšlenka fungování BTC

1. Transakce jsou přijímány uživateli.
2. Každý uživatel vybere nové transakce a vytvoří z nich nový blok B.
3. Jeden z uživatelů je vybrán náhodně a jeho blok B je přidán ke stávajícímu blockchainu. Na to potřebujeme konsenzus.
4. Nový blok bude odkazovat na B.

- Transakce
  - Pro provedení transakce je potřeba většinou znát předchozí. Jednou z výjimek je coinbase.
  - Syntax
    - Inputs: seznam hash-pointerů na předchozí transakce a podpis příjemce
    - Outputs: seznam adres příjemců a script
  - Script
    - Seznam instrukcí, jak příjemce dostane převáděné BTC.
    - Používá jednoduchý programovací jazyk, který je stack-based, bez cyklů. Ale i tak s ním lze napsat „smart-contracts“ jako například escrow.
    - ESCROW - Typ transakce zprostředkovávaný třetí stranou, které jsou zaslány peníze. Ty odešle druhé straně až po dodržení všech předem dohodnutých podmínek mezi odesilatelem a příjemcem.
  - Coinbase – typ transakce bez Inputs. Uživatel, který vytvořil blok, dostane BTC za nalezení a navíc poplatky od ostatních uživatelů za to, aby jejich transakce byla v bloku před ostatními.

- Mining
  1. Sesbíráme transakce a spočteme Merkel root.
  2. Vybereme hash posledního bloku v blockchainu.
  3. Spočteme nonce (proof of work) a spočteme hash našeho bloku. Pokud je validní, tak blok zašleme ostatním. Pokud ne zkusíme novou nonce.

  - Validita – je dán globální parametr d – difficulty. Hash bloku interpretujeme jako desetinné číslo a to musí být menší než d.
  - Chceme, aby nový blok byl nalezen cca každých 10 minut. Parametr d je každých 2016 bloků přepočítán tak, aby tato podmínka byla splněna.

    $$d = \frac{10d}{\text{Čas potřebný na nalezení posledních 2016 bloků}}$$

    Každý blok obsahuje timestamp, aby se tento čas dal zjistit.
  - Blokchain lze najít na `blockchain.info`

- Security
  - Není anonymní – každý uživatel má daný veřejný klíč a navíc všechny provedené transakce, lze najít veřejně v blockchainu.

- Double spending – použijeme stejné bitcoiny víckrát. BTC řeší tento problém tak, že příjemci čekají na potvrzení transakce. Transakce je potvrzená, pokud je dost bloků hluboko. $\Longrightarrow$ Pak je těžké bloky modifikovat.
- Selfish mining – miner neodešle hned svůj blok, ale rovnou začne pracovat na dalším. Část svých bloků odešle, až když se někdo další bude blížit dalšímu bloku. Dostane tak za nalezení bloků, víc než by měl.
- Pokud má adversary méně než $\frac{1}{3}$ výpočetního výkonu v síti, tak nemůže nic „zásadního“ provést. Zatím nedokázáno, není žádný rozumný model.

Nevýhody

- Sdružování uživatelů do poolů. Pokud se sdruží více uživatelů, kteří mají dohromady více jak polovinu výpočetní síly v systému, tak by mohli „ovládnout“ BTC.
- Spotřeba energie.
- Škálovatelnost. Nelze provést dost transakcí dostatečně rychle, aby BTC mohl konkurovat systémům jako VISA. Zvětšení bloků není řešení, zvýšíme tak latenci.