

# **Bezbednost u distribuiranim sistemima i osnovi kriptografije**

# Bezbednost u distribuiranim sistemima

- **Bezbednost** je jedan od **najteže ostvarivih principa** pošto mora **prožimati ceo distribuirani sistem**. Jedna greška u projektovanju bezbednosnih komponenti može učiniti **sve bezbednosne mere beskorisnim**
- Bezbednost je usko povezana sa zavisnošću (engl. *dependability*), koja uključuje dostupnost, pouzdanost, sigurnost i održivost
- Ako je potrebno da imamo poverenje u distribuirani računarski sistem, onda su veoma važne osobine i **poverljivost** (engl. *confidentiality*) i **integritet** (engl. *integrity*)
  - **poverljivost** se odnosi na svojstvo računarskog sistema koji svoje informacije otkriva samo autorizovanim stranama
  - **integritet** je svojstvo da se izmene u dobrima (engl. *asset*) sistema mogu vršiti samo na autorizovani način. Drugim rečima, nedozvoljene izmene u bezbednim računarskim sistemima mogu se detektovati i oporaviti

# Polise i mehanizmi bezbednosti

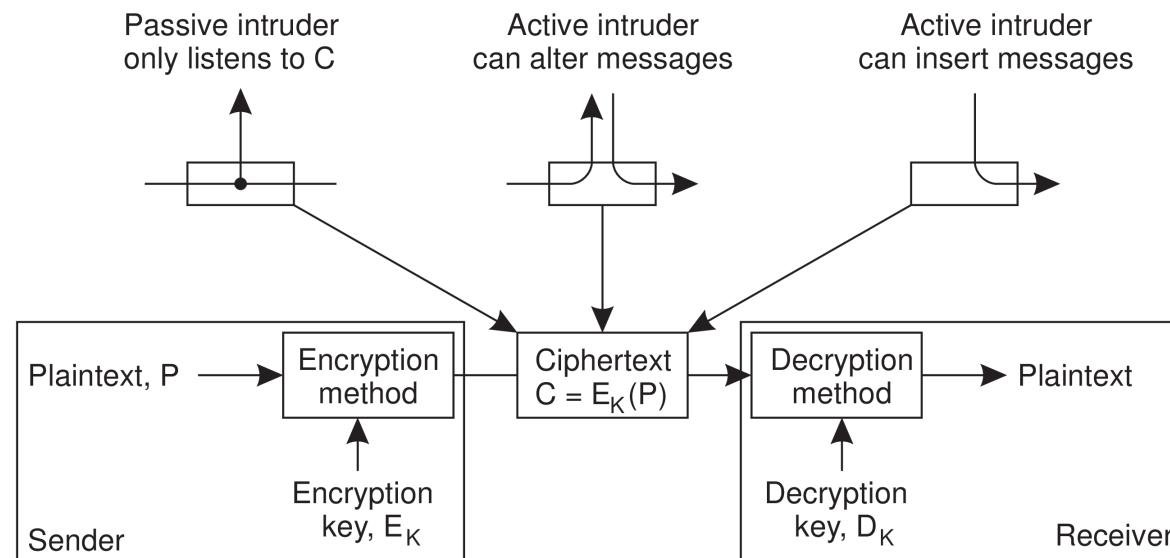
- Bezbedan računarski sistem se ne može izgraditi prostom tvrdnjom da treba biti u mogućnosti da se zaštiti od svih mogućih bezbednosnih pretnji
- Prvo je potreban **opis bezbednosnih zahteva** koji se naziva **polisa (politika) bezbednosti** (engl. security policy). **Polisa bezbednosti** precizno opisuje koje **akcije entiteti u sistemu smeju, odnosno ne smeju preuzeti**. Entiteti uključuju korisnike, servise, podatke, mašine, itd.
- Kada se postavi polisa bezbednosti, mogu se kreirati **mehanizmi bezbednosti** (engl. security mechanisms) pomoću kojih se polisa može sprovesti. Važni mehanizmi bezbednosti su:
  1. **šifrovanje** (engl. **encryption**)
  2. **autentifikacija** (engl. **authentication**), tj. provera identiteta
  3. **autorizacija** (engl. **authorization**)
  4. **revizija** (engl. **auditing**)

# Mehanizmi bezbednosti

1. **Šifrovanje transformiše podatke u oblik koji napadači ne mogu razumeti**, tj. pruža način za implementaciju poverljivosti podataka. Šifrovanje dodatno omogućava proveru da li su podaci modifikovani, tj. proveru integriteta
2. **Autentifikacija** se koristi za **verifikaciju identiteta** korisnika, klijenta, servera ili drugog entiteta. U slučaju klijenta, osnovna premla je da pre nego što servis počne da radi bilo šta u ime klijenta, servis mora znati identitet klijenta. Tipično, korisnici se autentikuju putem lozinki (engl. *passwords*)
3. **Autorizacija** – nakon autentifikacije klijenta, neophodno je proveriti **da li je klijent autorizovan da izvrši traženu akciju**. Npr. zavisno od toga ko pristupa bazi podataka, dozvola može biti dana za čitanje zapisa, modifikaciju određenih polja u zapisu ili dodavanje ili brisanje celih zapisa
4. **Revizija** – alati za reviziju koriste se za **praćenje koji klijenti pristupaju čemu i na koji način**. Iako sama revizija ne pruža zaštitu od bezbednosnih pretnji, revizioni logovi mogu biti veoma korisni za analizu bezbednosnih propusta i preduzimanje mera protiv uljeza. Zbog toga se napadači trude da ne ostave nikakve tragove koji bi doveli do otkrivanja njihovog identiteta. U ovom smislu, postojanje logova čini napade na računarske sisteme riskantnijim

# Kriptografija

- **Kriptografija je umetnost i nauka čuvanja bezbednosti poruka** (Schneier)
- **Pošiljalac** (engl. sender) S želi da pošalje **poruku** (engl. message)  $m$ , **primalac** (engl. receiver) R. Kako bi zaštitio poruku od bezbedonosnih pretnji, S prvo **šifruje**  $m$  u nerazumljivu poruku  $m'$  i potom šalje  $m'$  primaocu R. R mora **dešifrovati** primljenu poruku u originalnu formu  $m$ . Šifrovanje i dešifrovanje se postižu primenom kriptografskih metoda parametrizovanih ključevima. Originalni oblik poruke naziva se **otvoren tekst** (engl. **plaintext**), šifrovana forma je **šifrat** (engl. **ciphertext**)



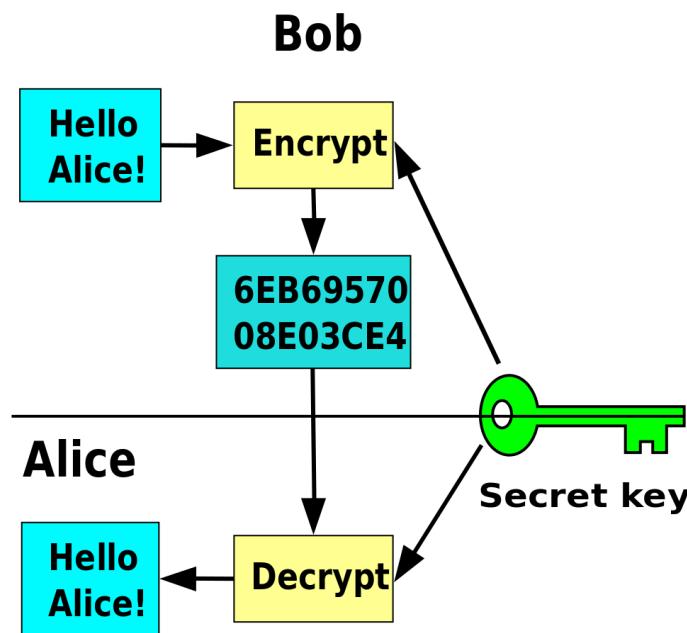
Izvor: <https://www.distributed-systems.net/index.php/books/distributed-systems-3rd-edition-2017>

# Simetrični i asimetrični kriptosistemi

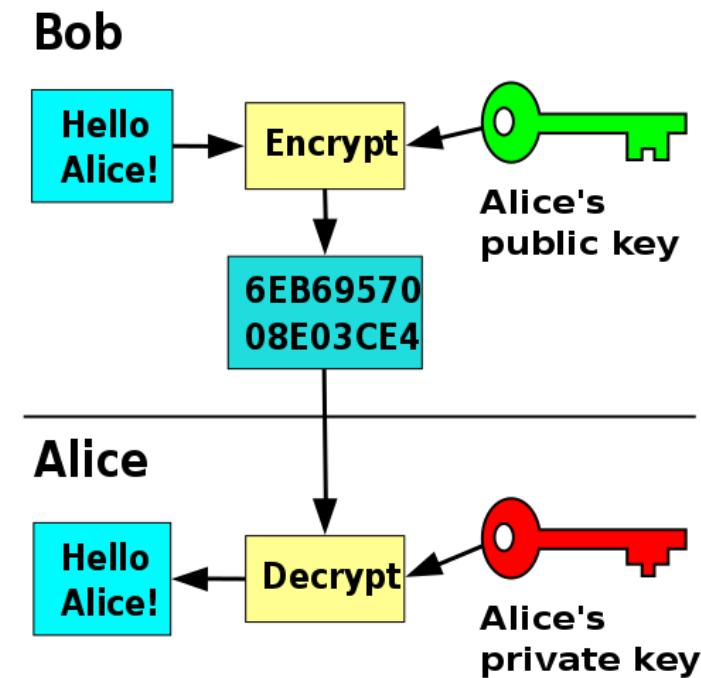
- U **simetričnom kriptosistemu**, isti ključ se koristi i za šifrovanje i za dešifrovanje poruke:  $P = D_k(E_k(P))$
- Simetrični kriptosistemi su poznati i kao **sistemi sa tajnim** (engl. *secret-key*) ili **deljenim ključem** (engl. *shared-key*), zato što se zahteva da posiljalac i primalac dele isti ključ, da bi se osiguralo da zaštita radi, **deljeni ključ** se mora održati **tajnim**, pa komunikacija ide isključivo kroz **bezbedne kanale** (engl. *secure channels*). Primeri: DES, AES
- U **asimetričnom kriptosistemu**, ključevi za šifrovanje i dešifrovanje se razlikuju, ali zajedno čine **jedinstveni par**. Drugim rečima, postoji poseban **ključ za šifrovanje**  $K_E$  i **dešifrovanje**  $K_D$ , tako da je  $P = D_{K_D}(E_{K_E}(P))$
- Jedan od ključeva u asimetričnom kriptosistemu se čuva kao privatni, dok se drugi objavljuje **javno**. S toga se asimetrični kriptosistemi nazivaju i **sistemi sa javnim ključem** (engl. *public-key systems*). Primeri: Diffie-Hellman, RSA, ECC, digitalni potpisi

# Simetrični i asimetrični kriptosistemi

(a) simetrični kriptosistem



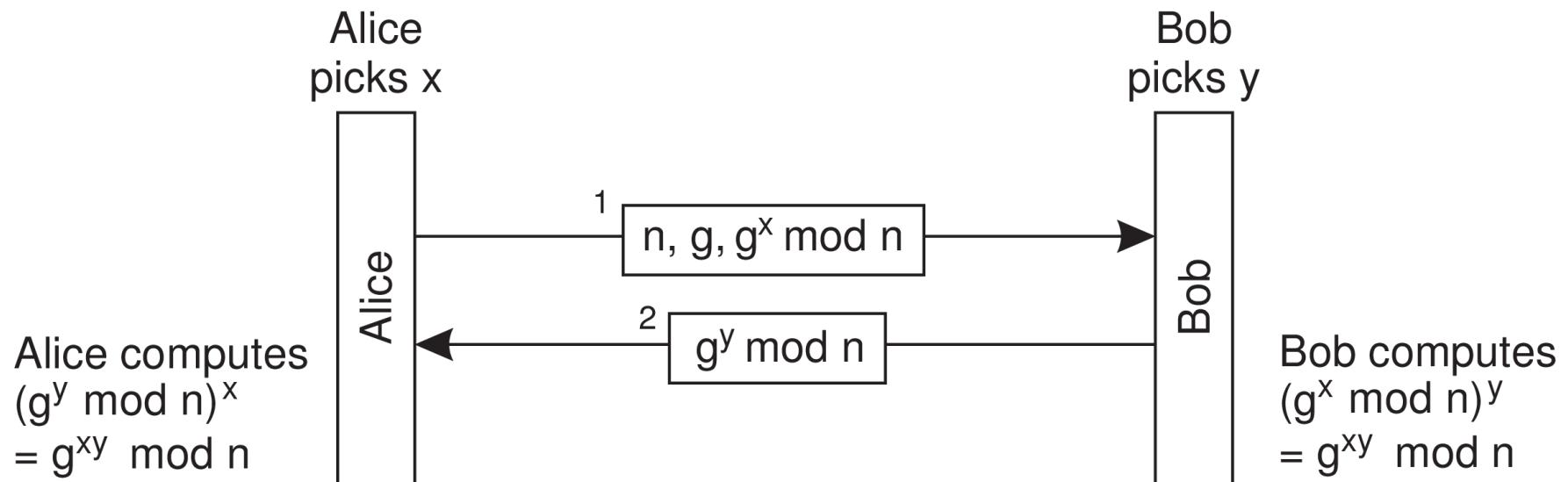
(b) asimetrični kriptosistem



Izvor: <https://en.wikipedia.org/wiki/Cryptography>

# Diffie-Hellman razmena ključeva

- Princip **Diffie-Hellman razmene ključeva**:
  - Alisa i Bob se dogovore o dva velika broja  $n$  i  $g$ , koji su oba javno poznati
  - Alisa bira slučajni broj  $x$  koji čuva kao tajnu (privatni ključ), Bob bira  $y$
  - Nakon koraka 1 i 2 sa slike, isključivo Alisa i Bob će imati tajni ključ  $g^{xy}$  mod  $n$



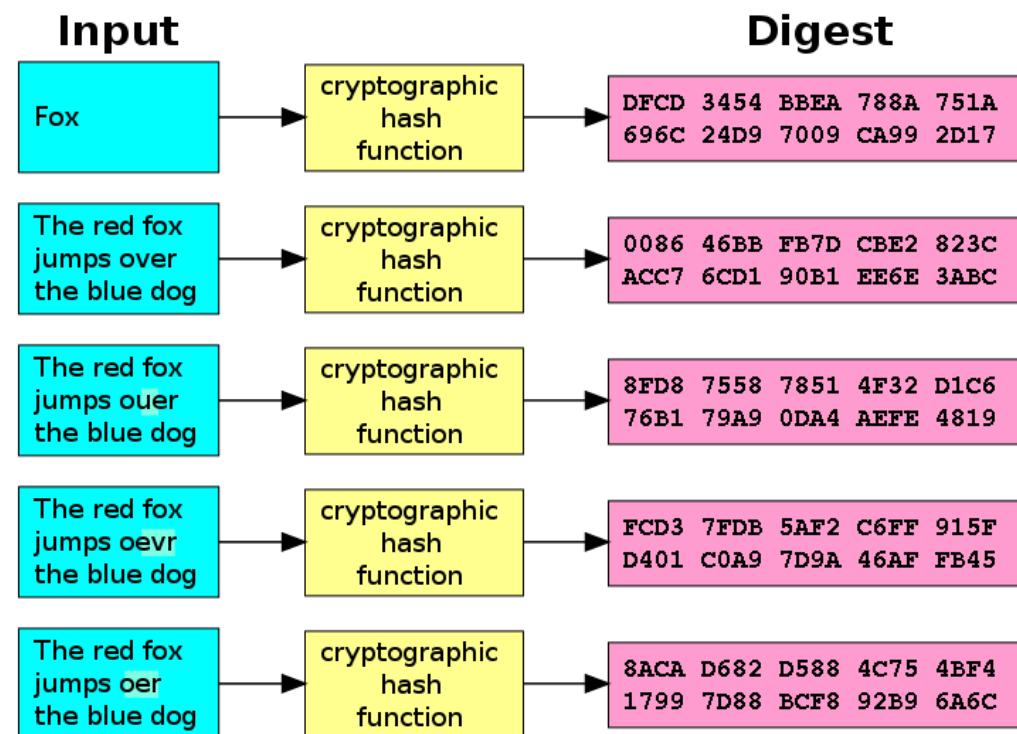
Izvor: <https://www.distributed-systems.net/index.php/books/distributed-systems-3rd-edition-2017>

# Kriptografske heš funkcije

- **Heš funkcija**  $H$  uzima kao ulaz poruku  $m$  proizvoljne dužine i proizvodi kao izlaz alfanumerički string  $h$  fiksne dužine:  $h = H(m)$
- Izlazni string se naziva **vrednost heša, sažetak poruke** (engl. *message digest*), **sažetak** ili **kontrolna suma** (engl. *checksum*)
- **Osobine idealne kriptografske heš funkcije:**
  - **jednosmernost** (engl. *one-way function*) označava da je neizvodljivo sa stanovišta složenosti izračunavanja (engl. *computationally infeasible*) pronaći ulaz  $m$  koji odgovara poznatom izlazu  $h$ . S druge strane, računanje  $h$  na osnovu  $m$  je lako. Schneier naziva jednosmerne heš funkcije „radnim konjima“ savremene kriptografije, primeri kriptografskih heš algoritama SHA-1/2/3, MD5
  - **slaba koliziona otpornost** (engl. *weak collision resistance*) označava da je, za dati ulaz  $m$  i njemu pridruženi izlaz  $h = H(m)$ , neizvodljivo sa stanovišta složenosti izračunavanja pronaći neki drugi ulaz  $m' \neq m$  tako da je  $H(m) = H(m')$
  - **jaka koliziona otpornost** (engl. *strong collision resistance*) označava da, kada je poznato samo  $H$ , nije izvodljivo sa stanovišta složenosti izračunavanja da se pronađu bilo koja dva različita ulaza  $m$  i  $m'$ , tako da je  $H(m) = H(m')$

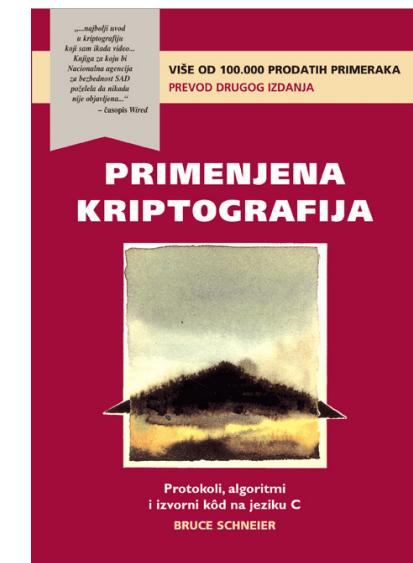
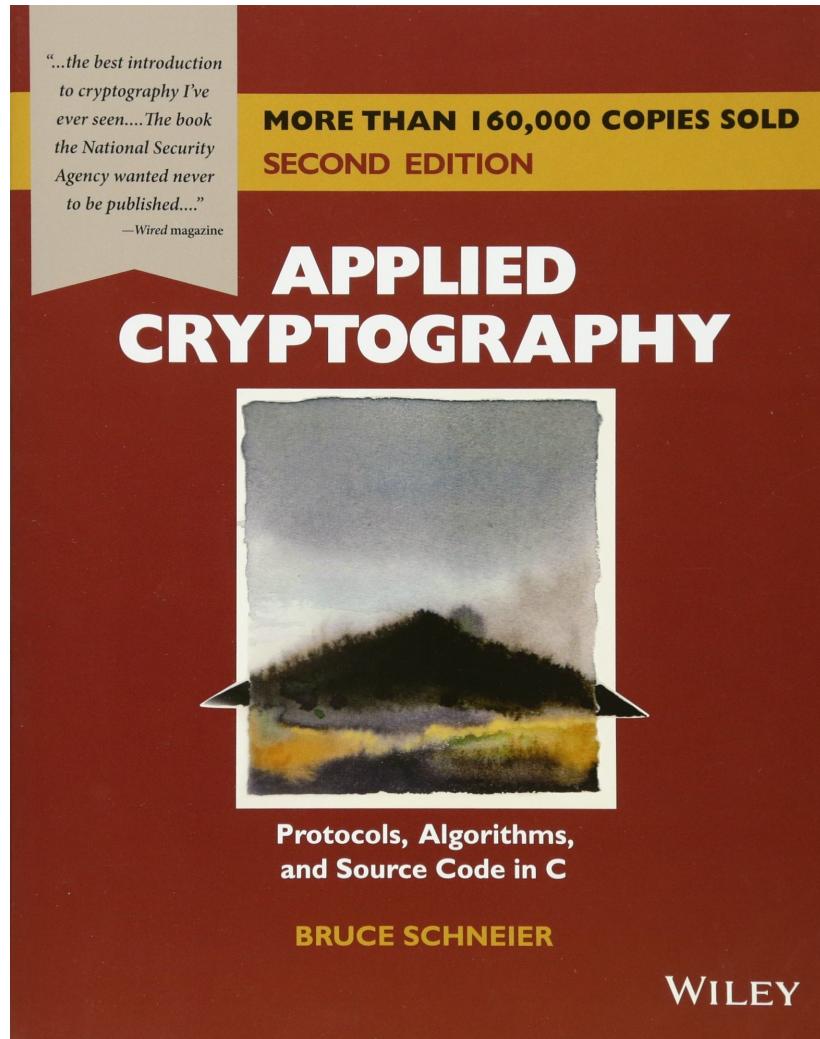
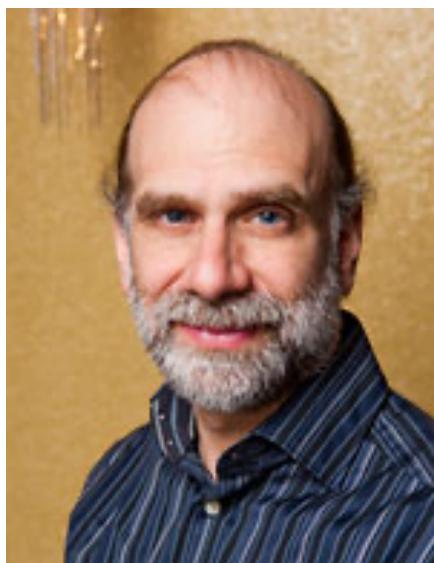
# Primer: kriptografske heš funkcije

- Čak i **male promene u ulaznim vrednostima** (u datom primeru kod reči "over") dovode do **drastičnih promena u rezultujućim izlazima**, tzv. **efekat lavine** (engl. *avalanche effect*):



Izvor: [https://simple.wikipedia.org/wiki/Cryptographic\\_hash\\_function](https://simple.wikipedia.org/wiki/Cryptographic_hash_function)

# Literatura



Izvor: [https://www.schneier.com/books/applied\\_cryptography/](https://www.schneier.com/books/applied_cryptography/)

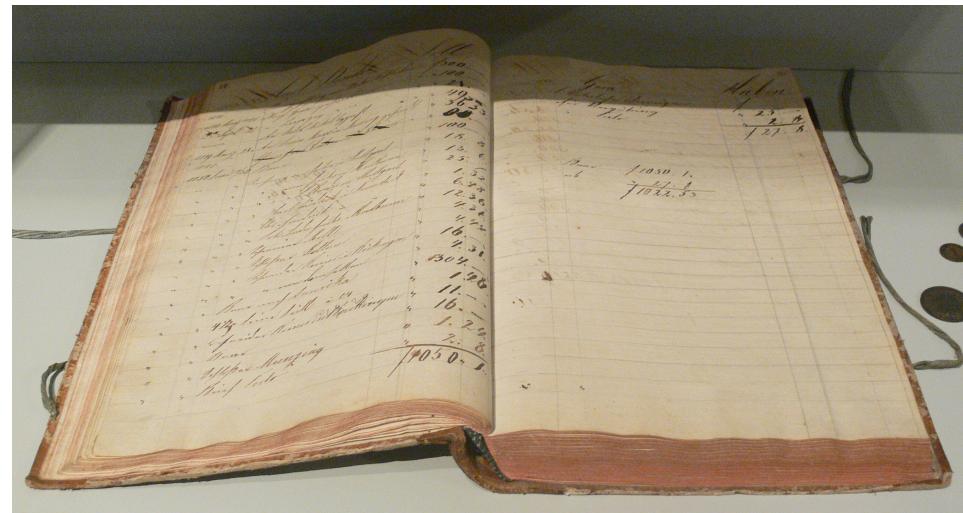
# Uvod u blokčejn

# Različita značenja termina

- **Sam Bitcoin, Ethereum i ostale kriptovalute**
- **Određena blokčejn tehnologija** koja pruža osnovu za rad Bitcoina i drugih kriptovaluta
- **Ideja blokčejna** kao novog načina za beleženje podataka o transakcijama

# Glavna knjiga

- Vođenje **računovodstvenih knjiga sa dva unosa** (engl. *double entry bookkeeping*)
- **Glavna knjiga** (engl. *ledger*) služi za beleženje transakcija, sa dugovanjima i potraživanjima u posebnim kolonama, kao i početnim i krajnjim stanjem računa
- Za transakciju se beleži - na teret kog računa (**debit**), u korist kog računa (**kredit**), kao i iznos



Izvor: [https://en.wikipedia.org/wiki/Ledger#/media/File:Hauptbuch\\_Hochstetter\\_vor\\_1828.jpg](https://en.wikipedia.org/wiki/Ledger#/media/File:Hauptbuch_Hochstetter_vor_1828.jpg)

# Distribuirana glavna knjiga

- **Distribuirana glavna knjiga** (engl. *distributed ledger*) ili **tehnologija distribuirane glavne knjige** (engl. *distributed ledger technology*) je način implementacije glavne knjige kod koga više nezavisnih “knjigovođa” u svojim kopijama beleže sve validne transakcije
- Usled postojanja više knjigovođa, neophodno je unapred dogovoriti:
  - **pravila** po kojima se utvrđuje koje **transakcije** su **validne**, kao i
  - **mehanizam postizanja dogovora (konsenzusa)** o tome koje će transakcije i u kom redosledu biti upisane
  - način na koji će se **razmenjivati poruke o transakcijama** između knjigovođa

# Primer: Distribuirana glavna knjiga

- Stanje glavne knjige:

Aca:  $A = 5, B = 2, C = 3$

Branka:  $A = 5, B = 2, C = 3$

Cveta:  $A = 5, B = 2, C = 3$

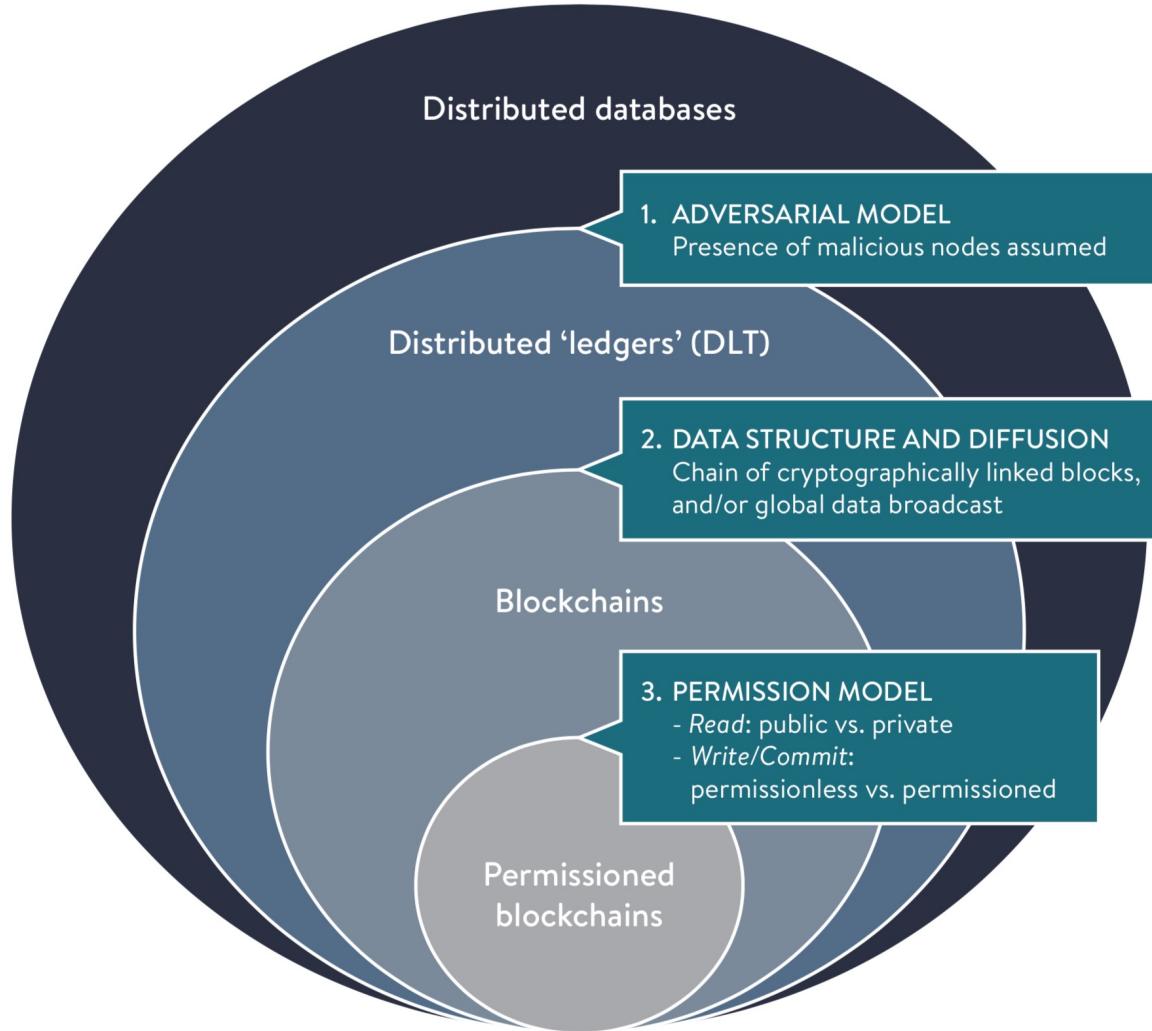
- Aca predlaže transakciju: Aca šalje Branki 2 dinara
- Predložena transakacija stiže do svih u mreži. Svi proveraju:
  - Da li A ima dovoljno sredstava za predloženu transakciju?
  - Da li ista sredstva trenutno nisu neophodna i za neku drugu predloženu transakciju?
- Ako su dogovorena pravila zadovoljena, svako upisuje predloženu transakciju u svoju kopiju glavne knjige, novo stanje svih kopija glavne knjige:

$Aca = 3, Branka = 4, Cveta = 3$

# Distribuirana baza, glavna knjiga i blokčejn

- **Distribuirana baza podataka** (engl. *distributed database*) je vrsta baze podataka kod koje se **podaci čuvaju u više čvorova** (računara)
- **Distribuirana glavna knjiga** (engl. *distributed ledger*) ili **tehnologija distribuirane glavne knjige** (engl. *distributed ledger technology – DLT*) je vrsta distribuirane baze podataka koja prepostavlja moguće **prisustvo malicioznih korisnika** (čvorova), vrsta strukture podataka za pamćenje transakcija, razmeštena na više čvorova
- **Blokčejn** (engl. *blockchain*) je **distribuirana struktura podataka** koja implementira **distribuiranu glavnu knjigu**, a sastavljena je od **lanca kriptografski povezanih blokova** koji **sadrže grupe transakcija**. U opštem slučaju, vrši se emitovanje (engl. *broadcast*) svih podataka svim učesnicima u mreži
- **Tokenizacija** (engl. *tokenisation*) se odnosi na proces digitalnog predstavljanja postojećeg (off-chain) dobra (engl. *asset*) u DLT

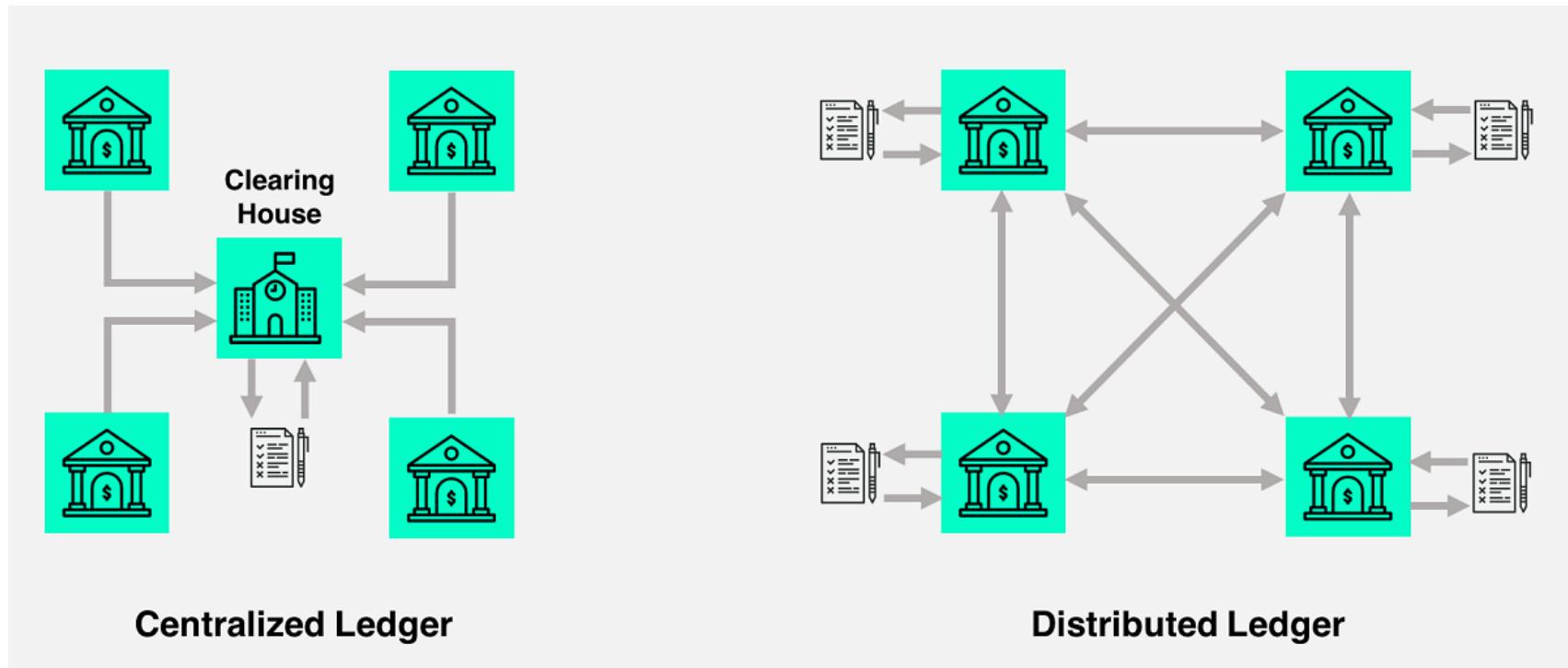
# Distribuirana baza, glavna knjiga i blokčejn



Izvor: [https://assets.ey.com/content/dam/ey-sites/ey-com/en\\_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf](https://assets.ey.com/content/dam/ey-sites/ey-com/en_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf)

# Distribuirana glavna knjiga – DLT

- **Centralizovana i distribuirana glavna knjiga:**



Izvor: <https://tradeix.com/distributed-ledger-technology/>

# Distribuirana glavna knjiga – DLT

- **DLT tehnologija** sastoji se od **tri glavne komponente**:
  - **model podataka** (engl. *data model*) obuhvata trenutno stanje glavne knjige
  - **jezik transakcija** (engl. *language of transactions*) kojim se vrši promena stanja glavne knjige
  - **protokol** (engl. *protocol*) se koristi kako bi se među učesnicima u distribuiranom sistemu postigao konsenzus o tome koje će transakcije biti prihvачene i u kom redosledu upisane u glavnu knjigu
- **DLT tehnologija** je osnova za **novu generaciju transakcionih aplikacija** koje uspostavljaju **poverenje** (engl. *trust*), **odgovornost** (engl. *accountability*) i **transparentnost** (engl. *transparency*), pri tom **racionalizujući poslovne procese i pravna ograničenja kroz automatizaciju**

# Distribuirana glavna knjiga – DLT

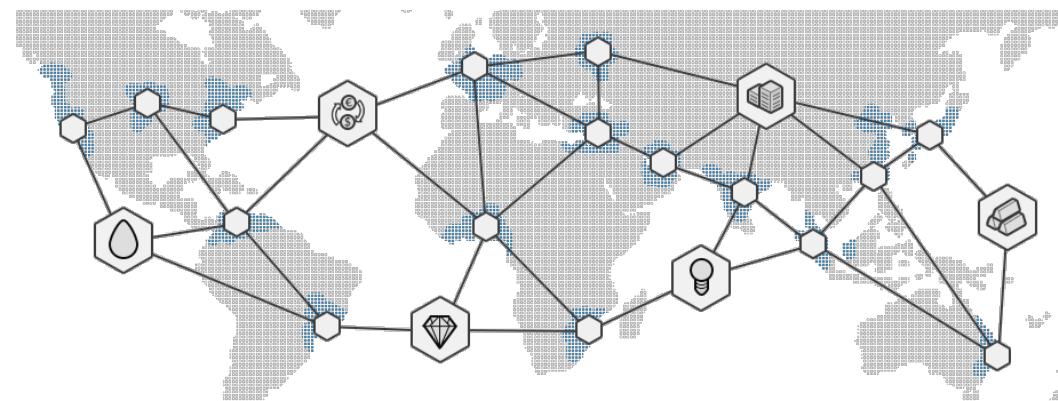
- Koncept DLT je postojao pre Bitcoina i blokčejn tehnologije. **Problem vizantijskih generala** (Lamport, Shostak i Pease, 1982), opisuje kako "računarski sistemi moraju da se nose sa suprotstavljenim informacijama" u neprijateljskom (engl. *adversarial*) okruženju
- Dalja istraživanja dovela su do **prvog algoritma (PBFT)** za **visoko dostupne sisteme koji mogu da tolerišu vizantijske otkaze** sa malim povećanjem latencije (Castro i Liskov, 1999)
- Najranije identifikovano pojavljivanje koncepta blokčejna je u radovima Haber i Stornetta 1991. (<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.46.8740>), kao i Bayer, Haber i Stornetta 1992. (<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.71.4891>), koji su uveli pojam lanca kriptografski povezanih blokova podataka kako bi se efikasno i bezbedno postavljali vremenski otisci na digitalne podatke u distribuiranim sistemima korišćenjem kriptografskih heš funkcija i Merkleovih stabala (engl. *Merkle trees*)
- **Bitcoin blokčejn** je 2008. doveo do **konvergencije skupa tehnologija**, uključujući **vremenski otisak transakcija, P2P mreže, kriptografiju** (digitalne potpise), **deljenu moć izračunavanja**, zajedno sa **novim konsenzus algoritmom**

# Blokčejn

- Glavna razlika između blokčejna i drugih distribuiranih baza podataka je u tome što je blokčejn projektovan kako bi se **postigao konzistentan i pouzdan dogovor o zapisu dogadaja** (npr. "ko je vlasnik čega") između **nezavisnih učesnika** koji mogu imati **različite motivacije i ciljeve**
- **Učesnici u blokčejn mreži postižu konsenzus o promenama stanja deljene baze podataka** (tj. transakcijama između učesnika) **bez potrebe da se veruje u integritet bilo kog učesnika mreže ili administratora**
- **Dogovor o stanju baze podataka** između učesnika u blokčejn mreži postiže se **mehanizmom konsenzusa**, koji osigurava da pogled na deljenu bazu podataka bude isti za svakog učesnika

# Blokčejn

- **Kombinacija konsenzus mehanizma sa specifičnom strukturu podataka** omogućava da se primenom blokčejna reši tzv. **problem dvostrukе potrošnje** (engl. *double spending problem*) – isti digitalni fajl se kopira i prenosi više puta – **bez zahteva za centralnom glavnom knjigom ili stranom koja bi sprečavala korisnike od dupliciranja ili potrošnje istog digitalnog fajla više puta**
- **Blokčejn** se, iz prethodnih razloga, može koristiti za **upravljanje prenosom dobara** (engl. *assets*) ili **drugih podataka bez potrebe za centralizovanim autoritetom** u koga svi moraju verovati



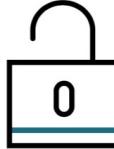
# Blokčejn

- **Blokčejn pruža mehanizme za:**
  - **Spajanje transakcija u blokove i njihovo beleženje**
  - **Kriptografsko povezivanje blokova u hronološkom redosledu**
  - **Održavanje i pristup kopijama glavne knjige**
- Bitcoin kao **prva primena blokčejna je rešio problem štampanja valute i dvostrukog potrošnje u digitalnom domenu**

# Komponente blokčejn tehnologije

## 1. Kriptografija

- kriptografske heš funkcije
- Merkleova stabla
- sistem sa javnim ključem



### CRYPTOGRAPHY

Use of a variety of cryptographic techniques including cryptographic one-way hash functions, Merkle trees and public key infrastructure (private-public key pairs)

## 2. P2P mreža

- javna ili privatna



### P2P NETWORK

Network for peer discovery and data sharing in a peer-to-peer fashion

## 3. Konsenzus mehanizam

- PoW, PoS, PoET,...
- PBFT, SBFT



### CONSENSUS MECHANISM

Algorithm that determines the ordering of transactions in an adversarial environment (i.e., assuming not every participant is honest)

## 4. Glavna knjiga

- lanac kriptografski povezanih blokova



### LEDGER

List of transactions bundled together in cryptographically linked ‘blocks’

## 5. Pravila važenja

- kako se ažurira glavna knjiga, koje transakcije su validne, itd.

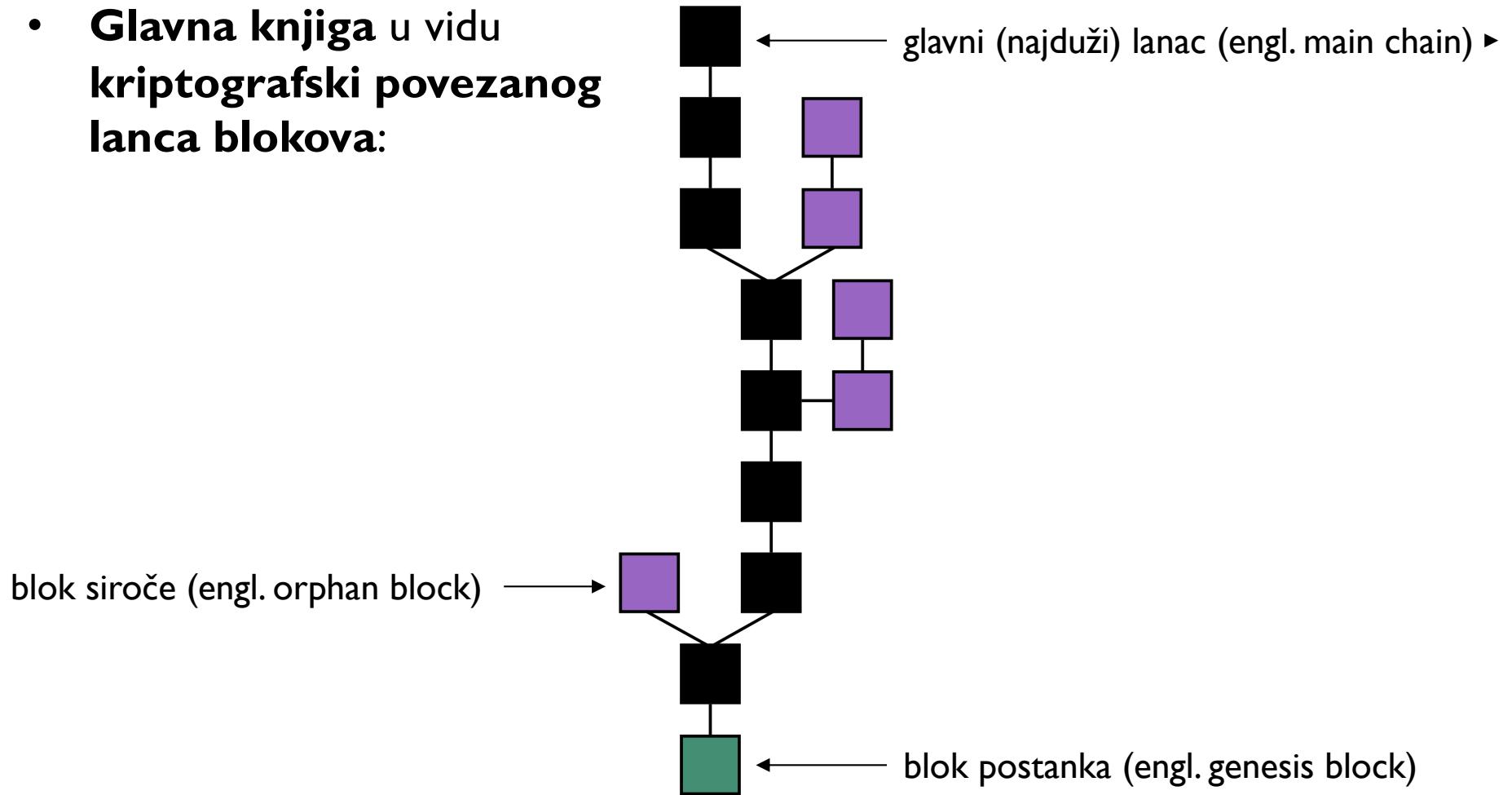


### VALIDITY RULES

Common set of rules of the network (i.e., what transactions are considered valid, how the ledger gets updated, etc.)

# Lanac blokova

- **Glavna knjiga u vidu kriptografski povezanog lanca blokova:**



Izvor: <https://en.wikipedia.org/wiki/Blockchain>

# Blok

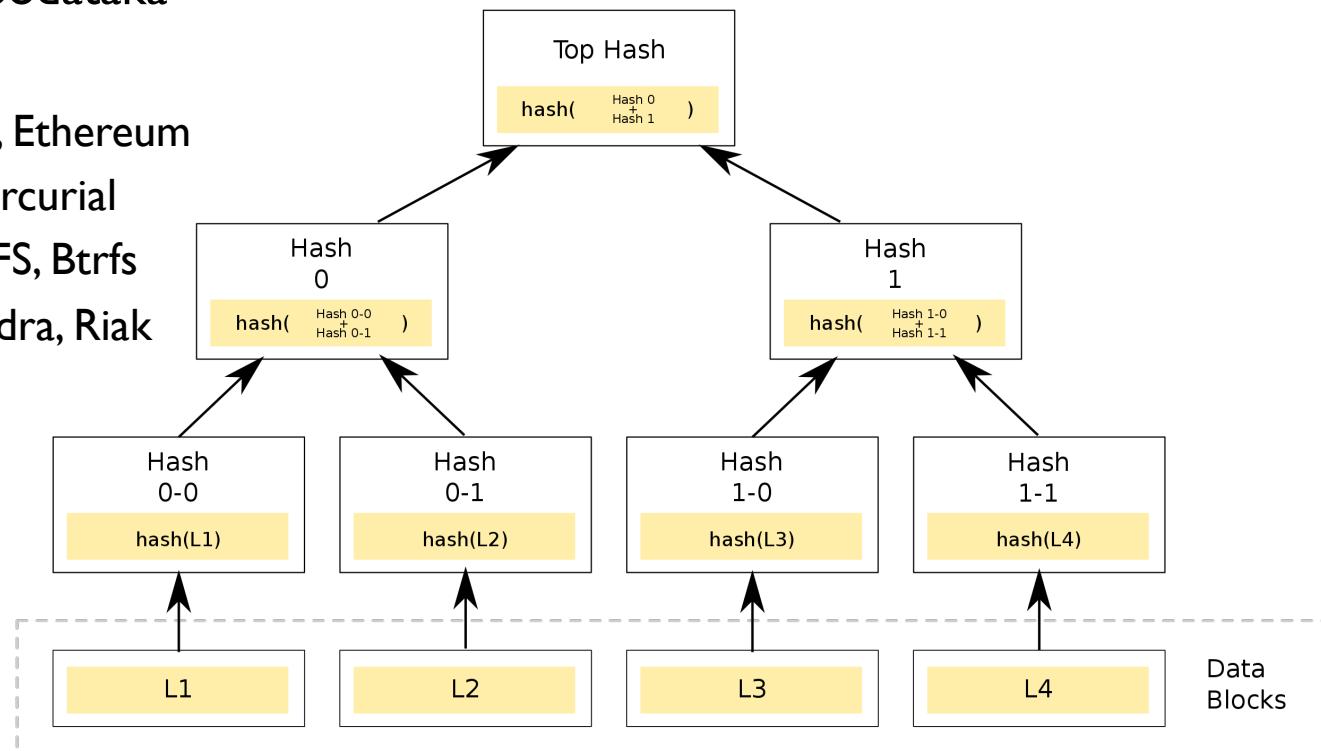
- **Blok** (engl. *block*) je skup transakcija koje se u kompletu istovremeno dodaju u lanac. Svaki blok sadrži određeni broj transakcija
- **Postavljanje vremenskog otiska** (engl. *timestamping*) je ključno svojstvo blokčejn tehnologije. Svaki blok sadrži vremenski otisak i svaki novi blok se referencira na prethodni. Zajedno sa **kriptografskim heševima**, ovakav lanac blokova sa vremenskim otiscima pruža neizmenjiv zapis svih transakcija u mreži, počev od prvog (tj. *genesis*) bloka
- Blok, tj. **zaglavlje** (engl. *header*) bloka, tipično sadrži **četiri metapodataka**:
  - **referencu na prethodni blok** u vidu **kriptografskog heša**
  - **nons** (engl. *nonce*), slučajni broj koji se koristi samo jedanput
  - **vremenski otisak**
  - **koren Merkleovog stabla** za transakcije uključene u blok

# Merkleovo stablo

- **Merkleovo stablo** (engl. Merkle tree – Ralph Merkle 1979.) ili **heš stablo** je stablo u kome je svaki **terminalni čvor** (list) označen **hešom bloka podataka**, a svaki **neterminalni čvor** je označen **kriptografskim hešom oznaka njegovih potomaka**. Heš stabla omogućavaju efikasnu i bezbednu verifikaciju sadržaja velikih struktura podataka

- **Primene:**

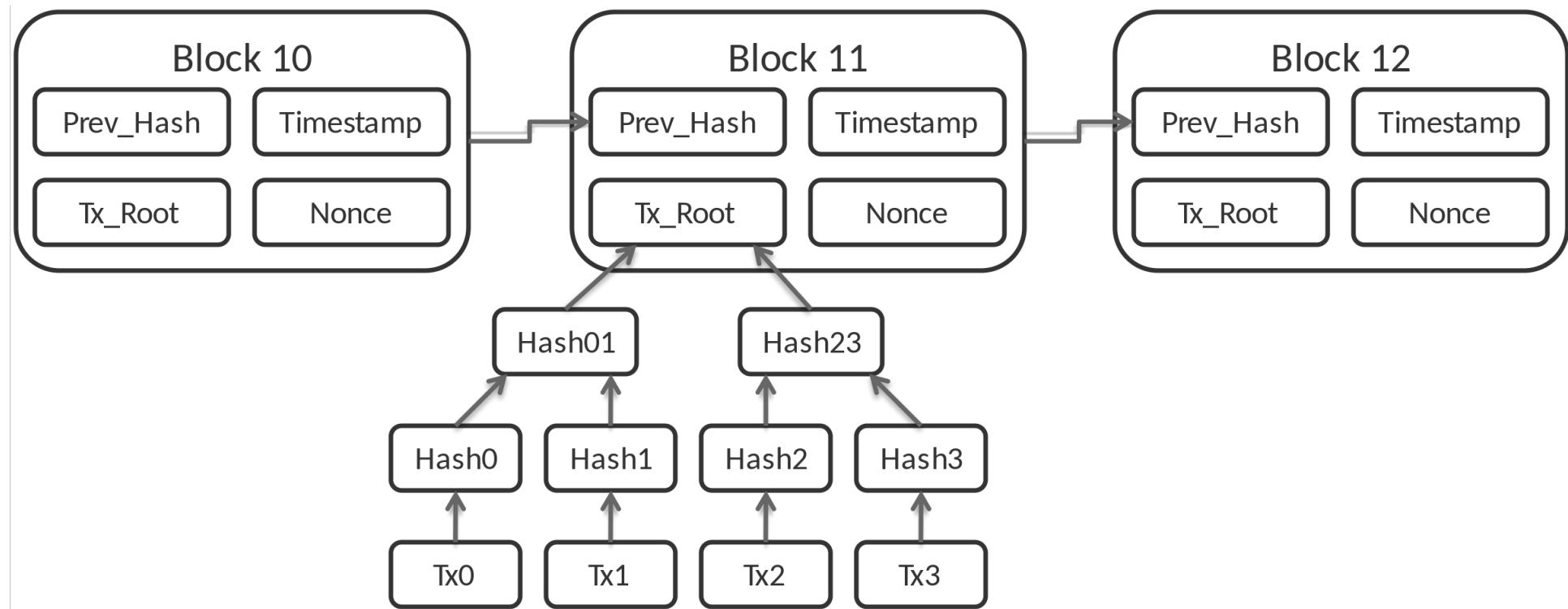
- Bitcoin, Ethereum
- Git, Mercurial
- IPFS, ZFS, Btrfs
- Cassandra, Riak



Izvor: [https://en.wikipedia.org/wiki/Merkle\\_tree](https://en.wikipedia.org/wiki/Merkle_tree)

# Primer: Bitcoin lanac blokova

- Primer globalne strukture Bitcoin lanca blokova:



Izvor: <https://en.wikipedia.org/wiki/Blockchain>

# Primer: Princip rada blokčejna

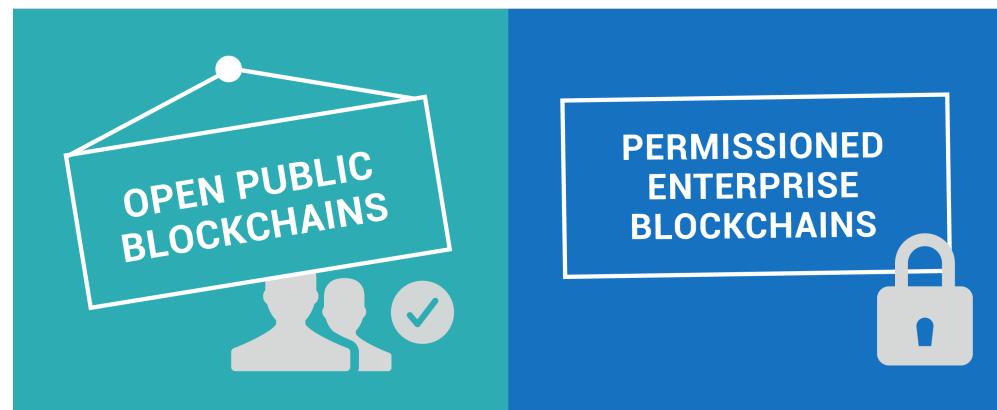
- Aca, Branka, Cveta i Dušan su učesnici u jednoj blokčejn mreži. Stanje računa (glavne knjige) je Aca = 5, Branka = 3, Cveta = 7, Dušan = 4.
- Dušan hoće da izvrši transakciju kojom šalje učesniku Cveti iznos od 3 BTC.
  - a) Ko mora da predloži tu transakciju i digitalno je potpiše?
  - b) Ko proverava da li je transakcija validna?
    - i) Šta moramo prvo proveriti?
    - ii) Da li je transakcija validna i ako Dušan u istom trenutku predlaže transakciju kojom šalje Aci iznos od 4 bitkoina? Kako vremenski otisak uz transakciju utiče na odluku po ovom pitanju?
  - c) Ko upisuje blokove sa validiranim transakcijama u svoju kopiju glavne knjige?

# Pametni ugovori

- **Pametni ugovor** (engl. *smart contract*) je **samo-izvršavajući računarski program** koji **automatski izvršava unapred definisane akcije** (npr. izvrši se plaćanje kada neki događaj okine (engl. *trigger*) pametni ugovor) kada se **određeni uslovi unutar sistema ispune**
- **Funkcionalnost pametnih ugovora** odnosi se na stepen funkcionalnosti DLT radnog okruženja ili mreže u smislu **složenosti izračunavanja** koja može **izvesti na lancu** (engl. *on-chain*)
  - **sa pamćenjem stanja** (engl. *stateful*) – **sekvencijalni**: logički-optimizovani sistem sa širokim funkcionalnostima pametnih ugovora na nivou protokola, pamti interno stanje, podržava iteracije i rekurziju, otežana verifikacija
  - **bez pamćenja stanja** (engl. *stateless*) – **kombinacioni**: transakciono-optimizovani sistem koji ne podržava složena izračunavanja na osnovnom nivou, nema internog stanja, lakša verifikacija

# Tipovi blokčejn mreža

- **Osnovna podela blokčejn P2P mreža na osnovu mogućnosti pristupa:**
  - **javne** (engl. *public*), **bez kontrole pristupa** (engl. *permissionless*), tj. sa slobodnim pristupom – bilo ko se može pridružiti mreži
  - **privatne** (engl. *private*), **sa kontrolom pristupa** (engl. *permissioned*), zahteva prethodnu verifikaciju učesnika u mreži koji su obično međusobno poznati
- **Izbor** između javnih i privatnih blokčejna **zavisi od slučaja korišćenja**



# Tipovi blokčejn mreža

- **Osnovna podela blokčejn mreža na osnovu mogućnosti pristupa:**

		Read	Write	Commit	Example	
Blockchain types	Open	Public permissionless	Open to anyone	Anyone	Anyone*	Bitcoin, Ethereum
	Open	Public permissioned	Open to anyone	Authorised participants	All or subset of authorised participants	Sovrin
	Closed	Consortium	Restricted to an authorised set of participants	Authorised participants	All or subset of authorised participants	Multiple banks operating a shared ledger
	Closed	Private permissioned ('enterprise')	Fully private or restricted to a limited set of authorised nodes	Network operator only	Network operator only	Internal bank ledger shared between parent company and subsidiaries

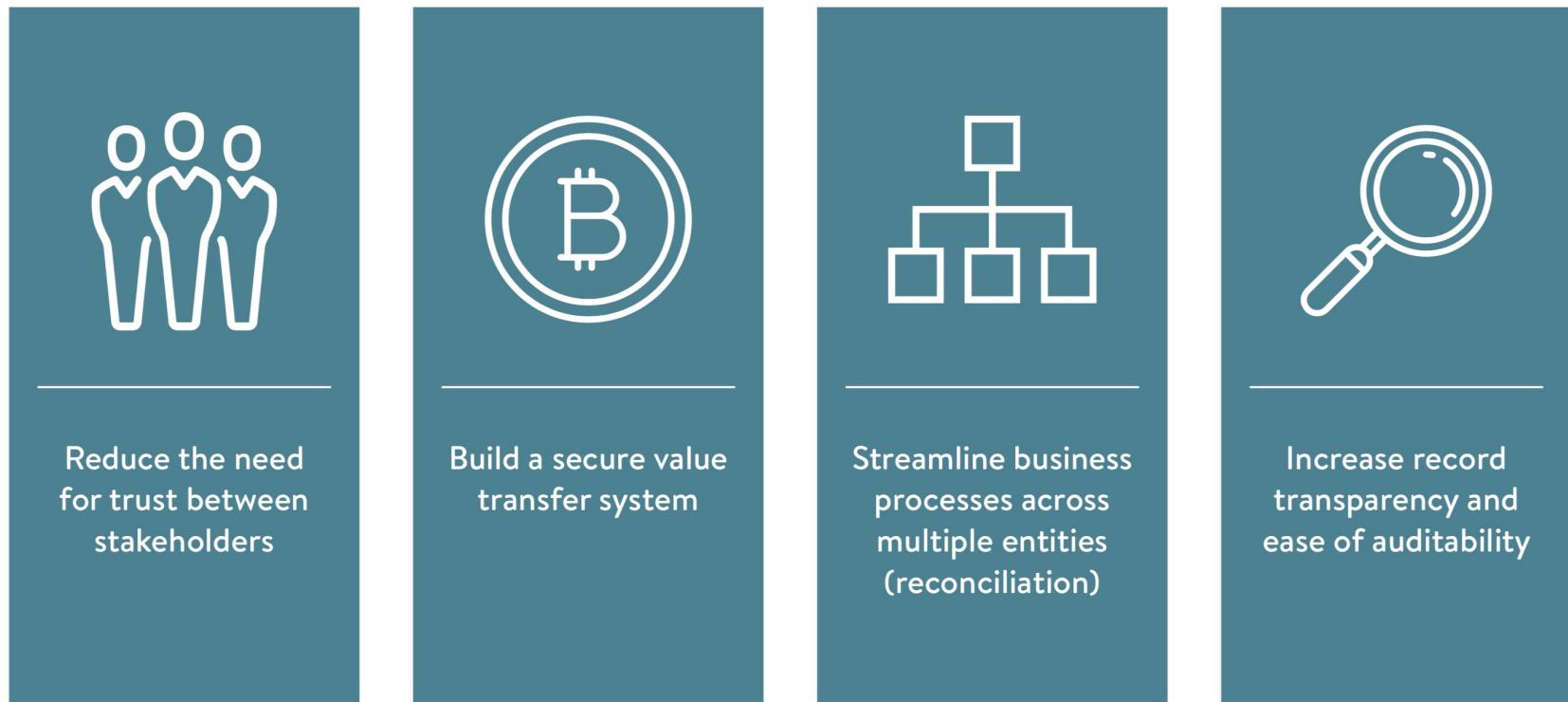
\* Requires significant investment either in mining hardware (proof-of-work model) or cryptocurrency itself (proof-of-stake model).

Izvor: [https://assets.ey.com/content/dam/ey-sites/ey-com/en\\_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf](https://assets.ey.com/content/dam/ey-sites/ey-com/en_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf)

# Primene blokčejna

*“Blockchains can help us advance from a ‘don’t be evil’ world to a ‘can’t be evil’ world.”*

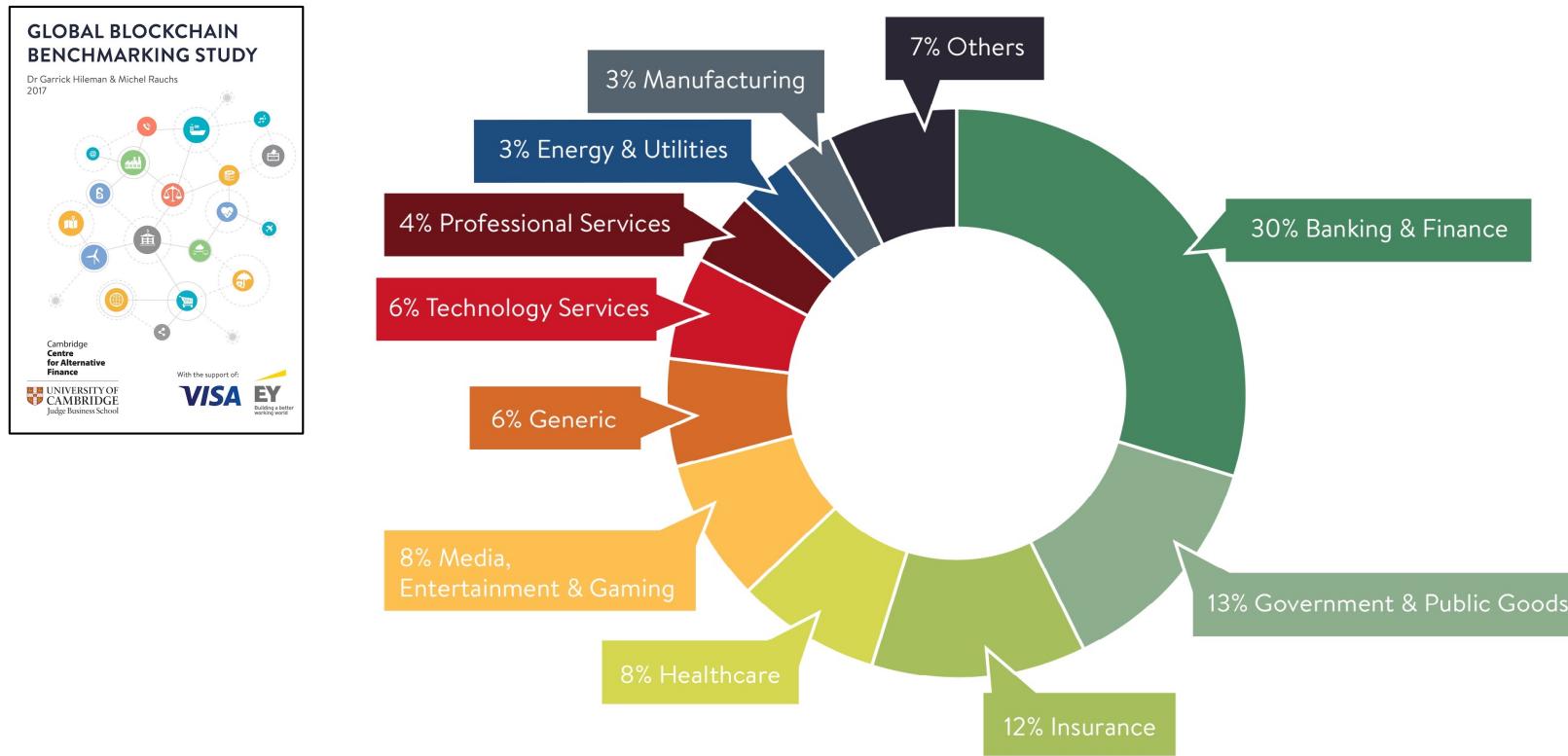
Muneeb Ali, Blockstack  
(<https://blockstack.org>)



Izvor: [https://assets.ey.com/content/dam/ey-sites/ey-com/en\\_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf](https://assets.ey.com/content/dam/ey-sites/ey-com/en_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf)

# Primene blokčejna

- Raspodela slučajeva korišćenja blokčejna prema granama privrede:



Note: This figure is based on a list of 132 use cases, grouped into industry segments, that have been frequently mentioned in public discussions, reports, and press releases.<sup>33</sup>

Izvor: [https://assets.ey.com/content/dam/ey-sites/ey-com/en\\_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf](https://assets.ey.com/content/dam/ey-sites/ey-com/en_gl/topics/emeia-financial-services/ey-global-blockchain-benchmarking-study-2017.pdf)

# Mitovi o blokčejnu



## MYTH

Blockchains are  
'trustless'

## REALITY

**Blockchains always require some  
degree of trust**



## MYTH

Blockchains are  
immutable or  
'tamper-proof'

## REALITY

**Transactions on a blockchain  
network can be reversed by  
network participants under  
specific circumstances**



## MYTH

Blockchains are  
100% secure

## REALITY

**Blockchains are not  
automatically more secure  
than other systems**



## MYTH

Blockchains are  
'truth machines'

## REALITY

**GIGO ('garbage in, garbage out')  
applies to every blockchain that  
uses non-native digital assets  
and/or external data inputs**

# Prednosti blokčejna

- **Nepromenljiva i transparentna baza podataka za sve učesnike**
- **Nema posrednika** u realizaciji transakcija
- **Nema potrebe za poverenjem** u **centralni autoritet**
- **Otporan na maliciozne učesnike**
- **Nema jednu tačku otkaza (SPOF), unapređena bezbednost**