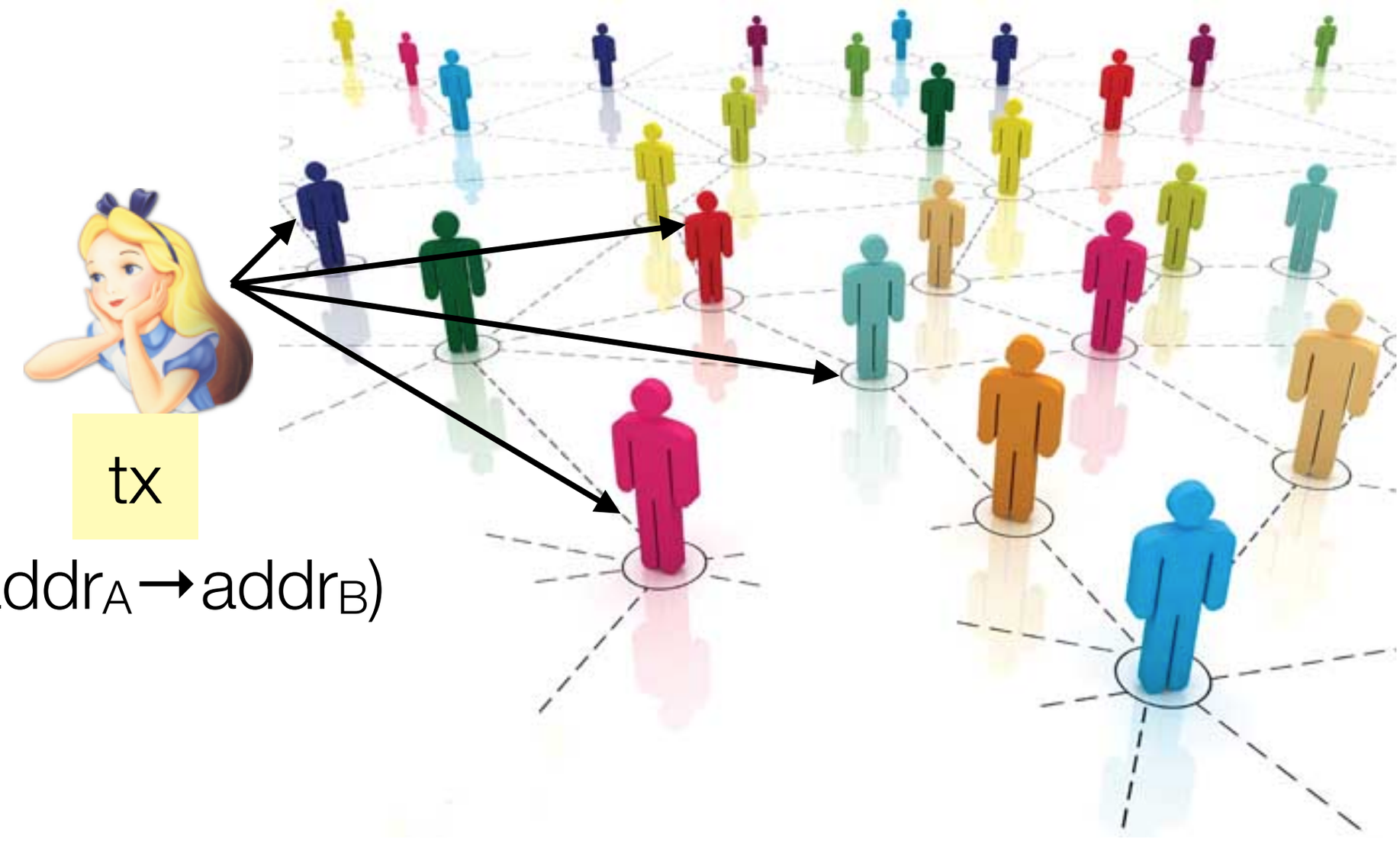
Alternatives to Blockchains

Sarah Meiklejohn (University College London)

fully decentralized cryptocurrencies

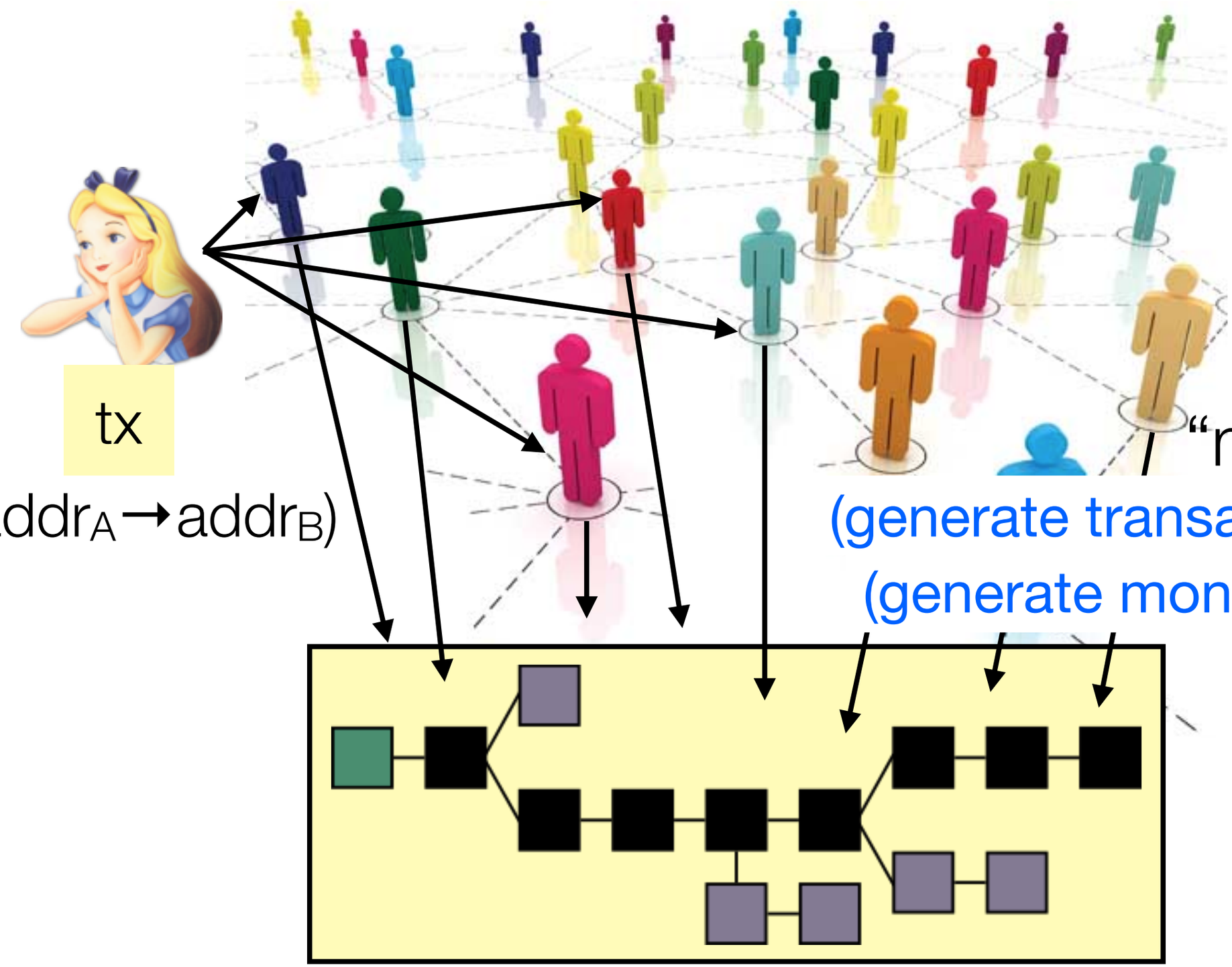
fully decentralized cryptocurrencies



tx

$tx(addr_A \rightarrow addr_B)$

fully decentralized cryptocurrencies



tx

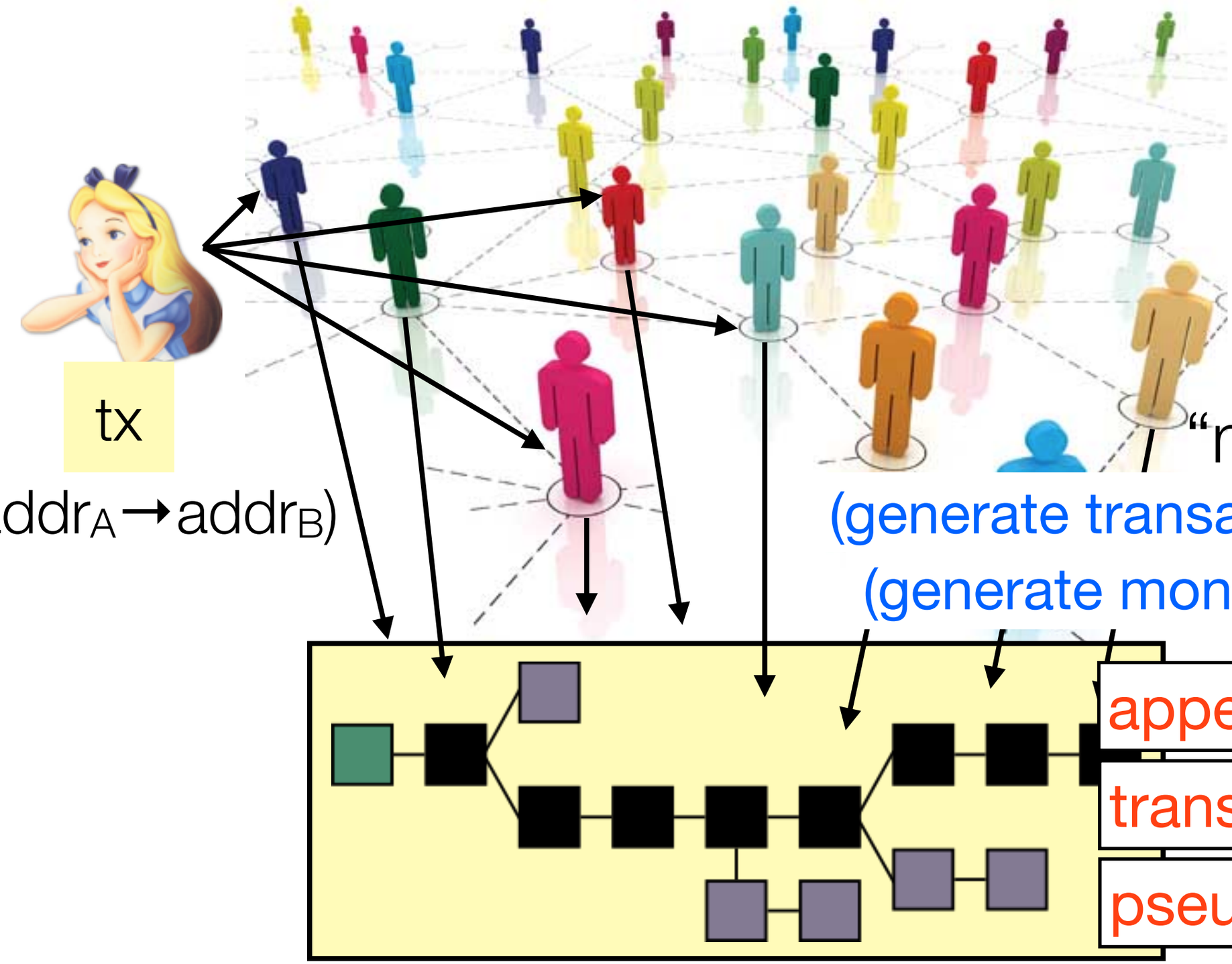
tx(addr$_A$→addr$_B$)

"mining"

(generate transaction ledger)

(generate monetary supply)

fully decentralized cryptocurrencies



tx

tx(addr$_A$→addr$_B$)

"mining"

(generate transaction ledger)

(generate monetary supply)

append-only

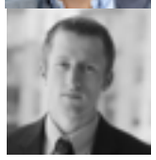
transparent

pseudonyms

issues with Bitcoin

lack of fungibility

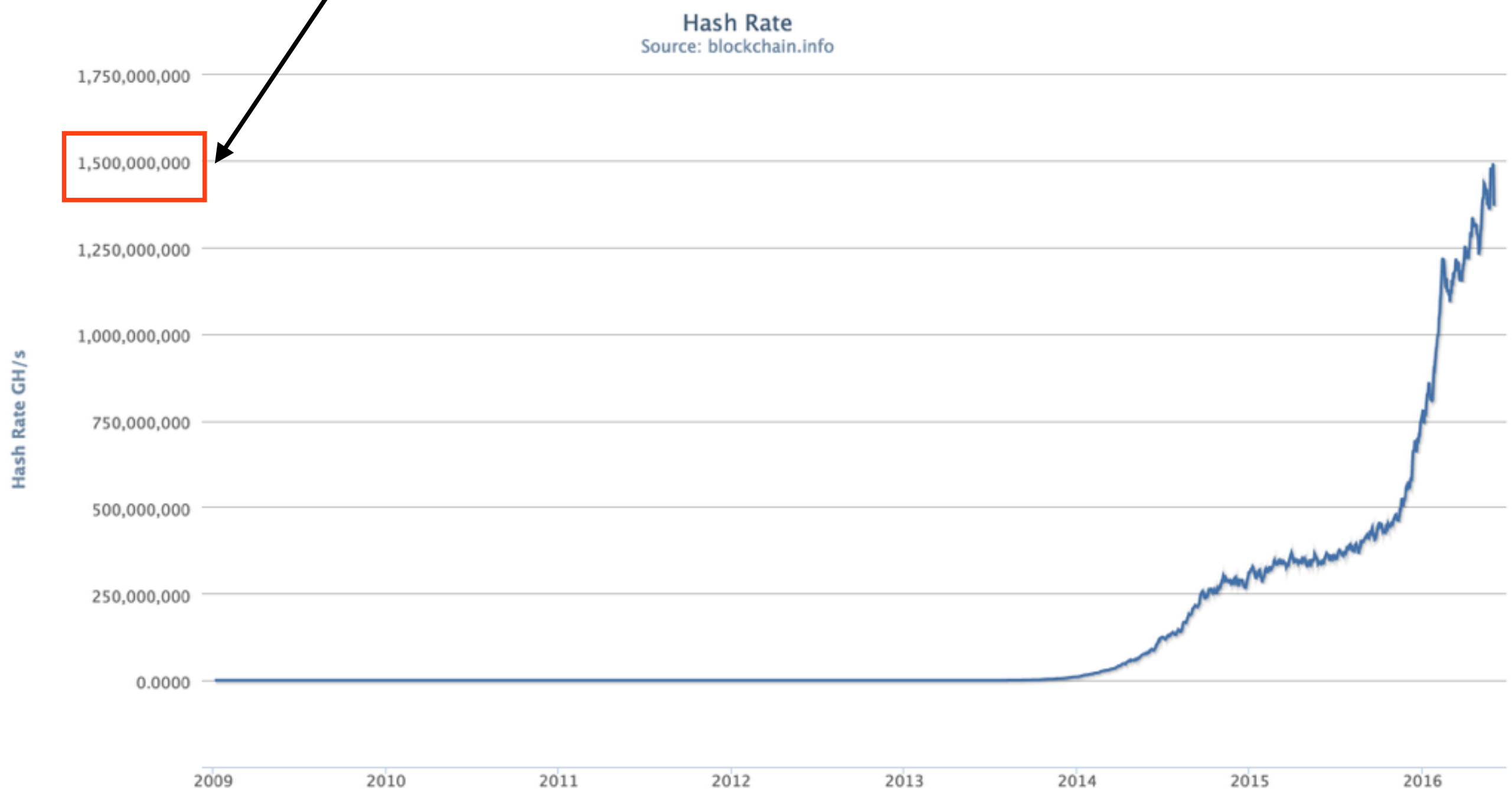
issues with Bitcoin



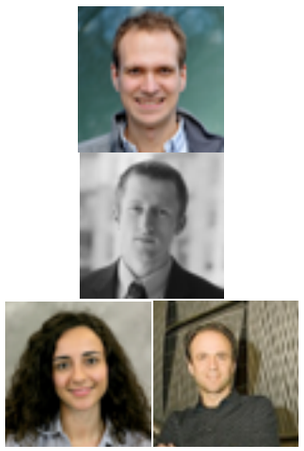
lack of fungibility

hashing rates are out of control

1.5 EH/s (1.5 x 10^{18} H/s)!



issues with Bitcoin

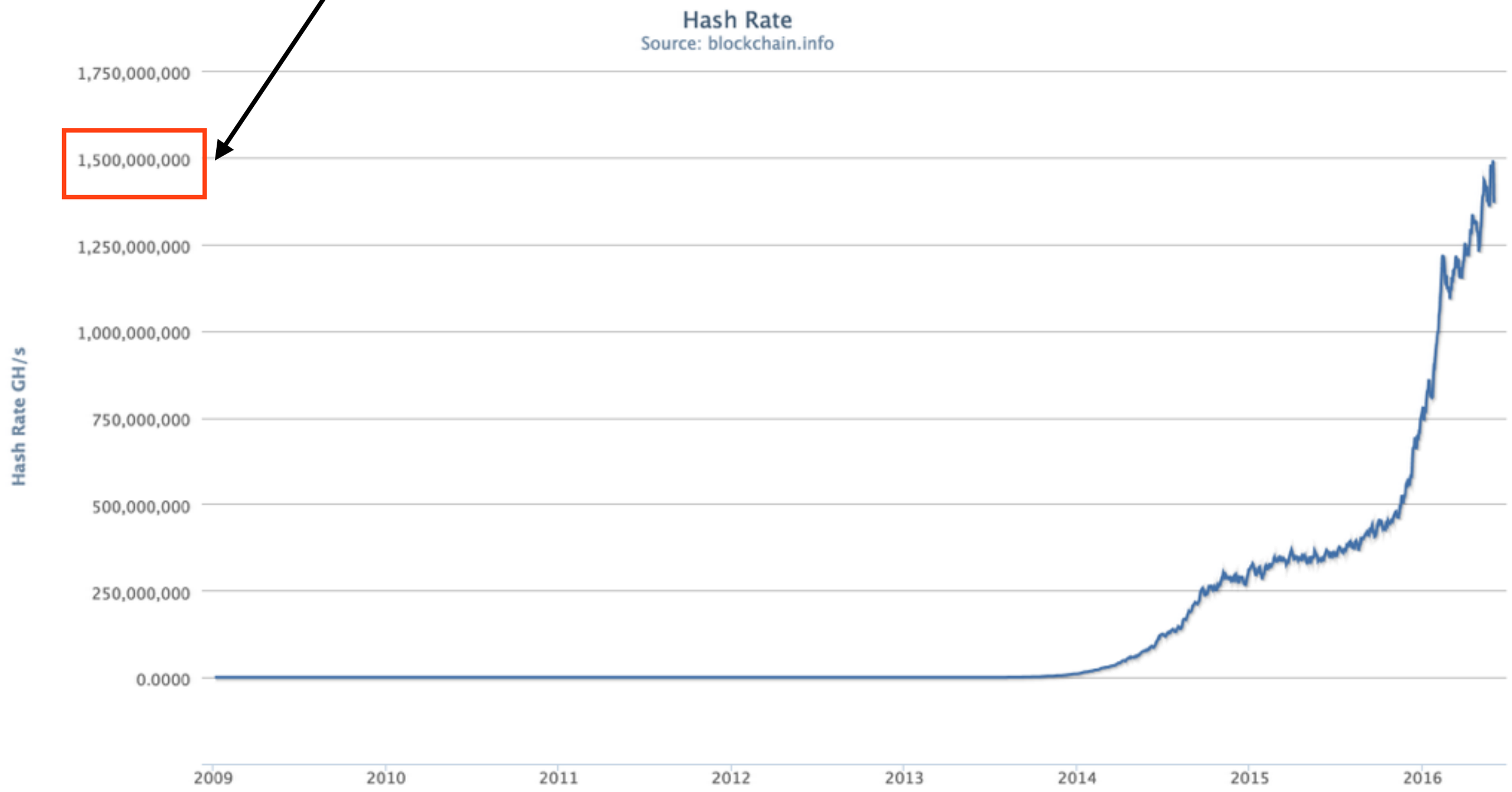


lack of fungibility

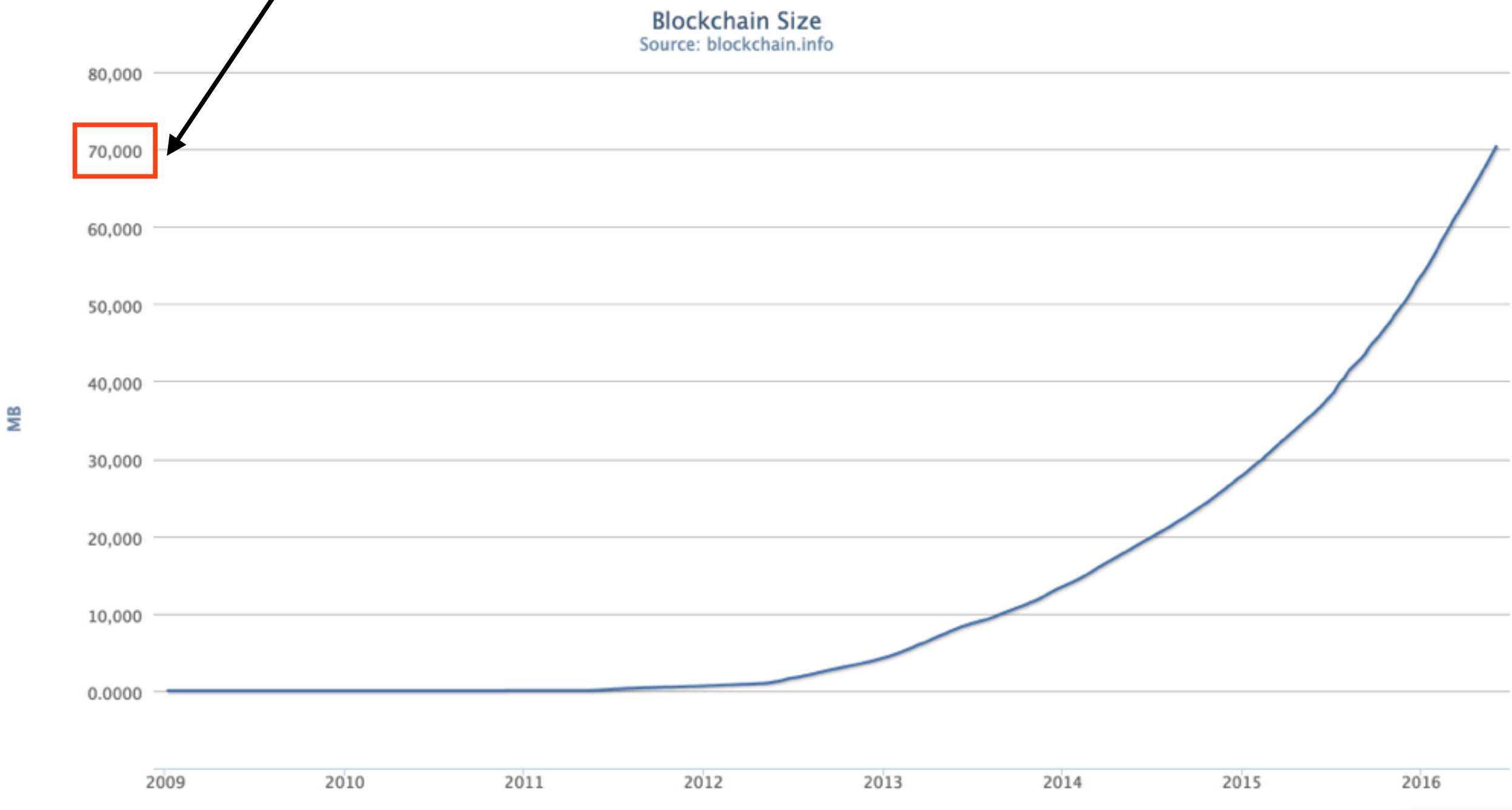
hashing rates are out of control

lack of scalability

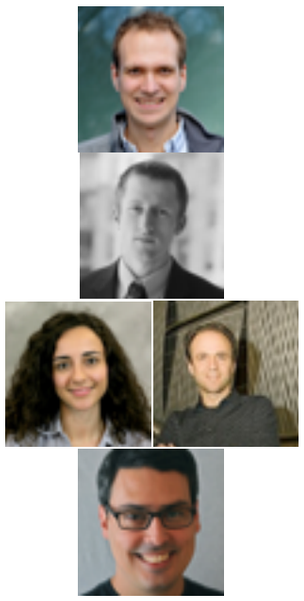
1.5 EH/s (1.5 x 10^{18} H/s) to achieve 7 tx/s!



70GB and no end in sight!



issues with Bitcoin



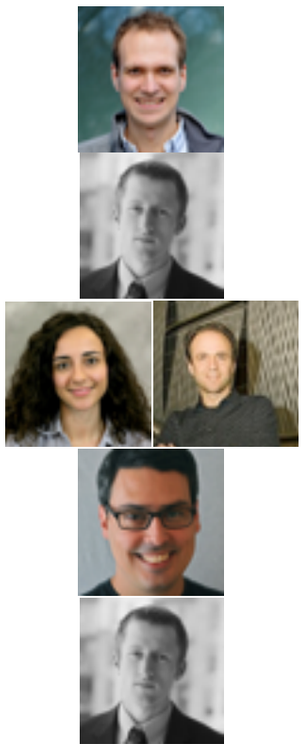
lack of fungibility

hashing rates are out of control

lack of scalability

lack of consumer protection

issues with Bitcoin



lack of fungibility

hashing rates are out of control

lack of scalability

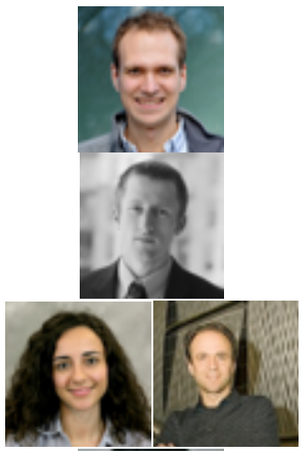
lack of consumer protection

attacks on mining / misaligned incentives

Majority is not Enough:
Bitcoin Mining is Vulnerable

Ittay Eyal and Emin Gün Sirer

issues with Bitcoin

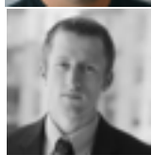


lack of fungibility

hashing rates are out of control

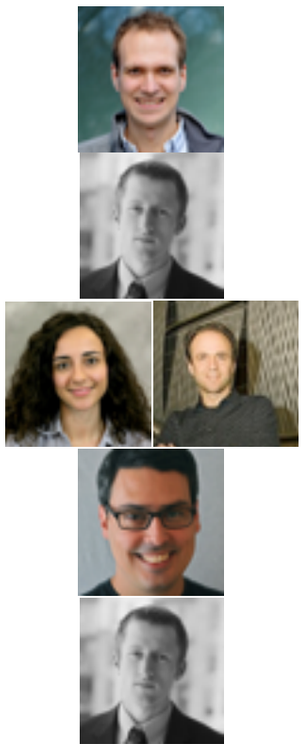
lack of scalability

lack of consumer protection



attacks on mining / misaligned incentives

issues with Bitcoin



lack of fungibility

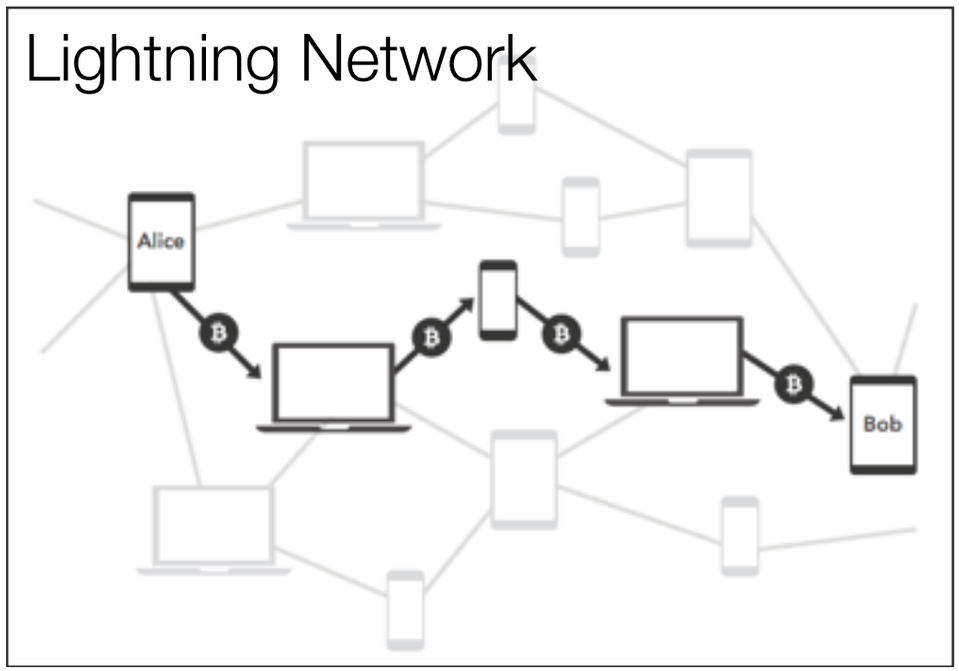
hashing rates are out of control

lack of scalability

lack of consumer protection

attacks on mining / misaligned incentives

alternate storage

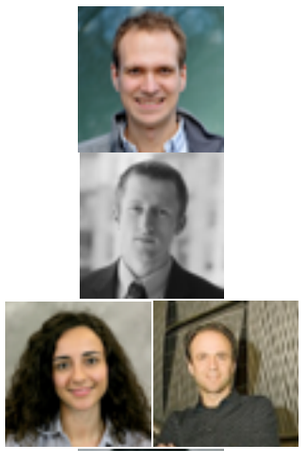


factom

truncate after a certain amount of time?

IPFS

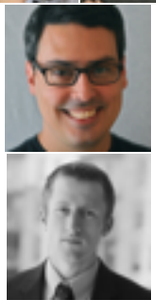
issues with Bitcoin



lack of fungibility

hashing rates are out of control

lack of scalability



lack of consumer protection

attacks on mining / misaligned incentives

alternate proof of …

Argon2: new generation of memory-hard functions for password hashing and other applications

Alex Biryukov
University of Luxembourg
alex.biryukov@uni.lu

Daniel Dinu
University of Luxembourg
dumitru-daniel.dinu@uni.lu

Dmitry Khovratovich
University of Luxembourg
khovratovich@gmail.com

Permacoin: Repurposing Bitcoin Work for Data Preservation

Andrew Miller[1], Ari Juels, Elaine Shi[1], Bryan Parno[2] and Jonathan Katz[1]

[1]University of Maryland
[2]Microsoft Research

SpaceMint: A Cryptocurrency Based on Proofs of Space*

Sunoo Park† Albert Kwon† Joël Alwen‡

Georg Fuchsbauer‡ Peter Gaži‡ Krzysztof Pietrzak‡

†*MIT* ‡*IST Austria*

proof of stake 

use of computational resources represents stake in system

other forms of stake?

(icon by lastspark for the noun project)

proof of stake

use of computational resources represents stake in system

other forms of stake?

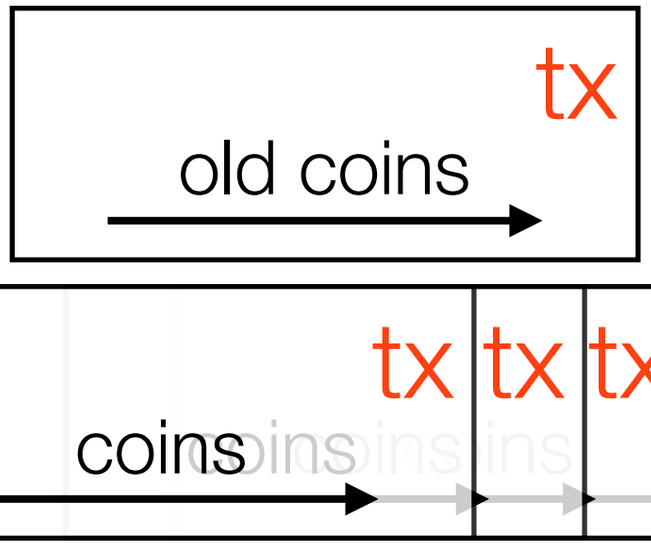
“proof of burn”
“proof of coin age”

(icon by lastspark for the noun project)

proof of stake

use of computational resources represents stake in system

other forms of stake?



“proof of burn”
“proof of coin age”

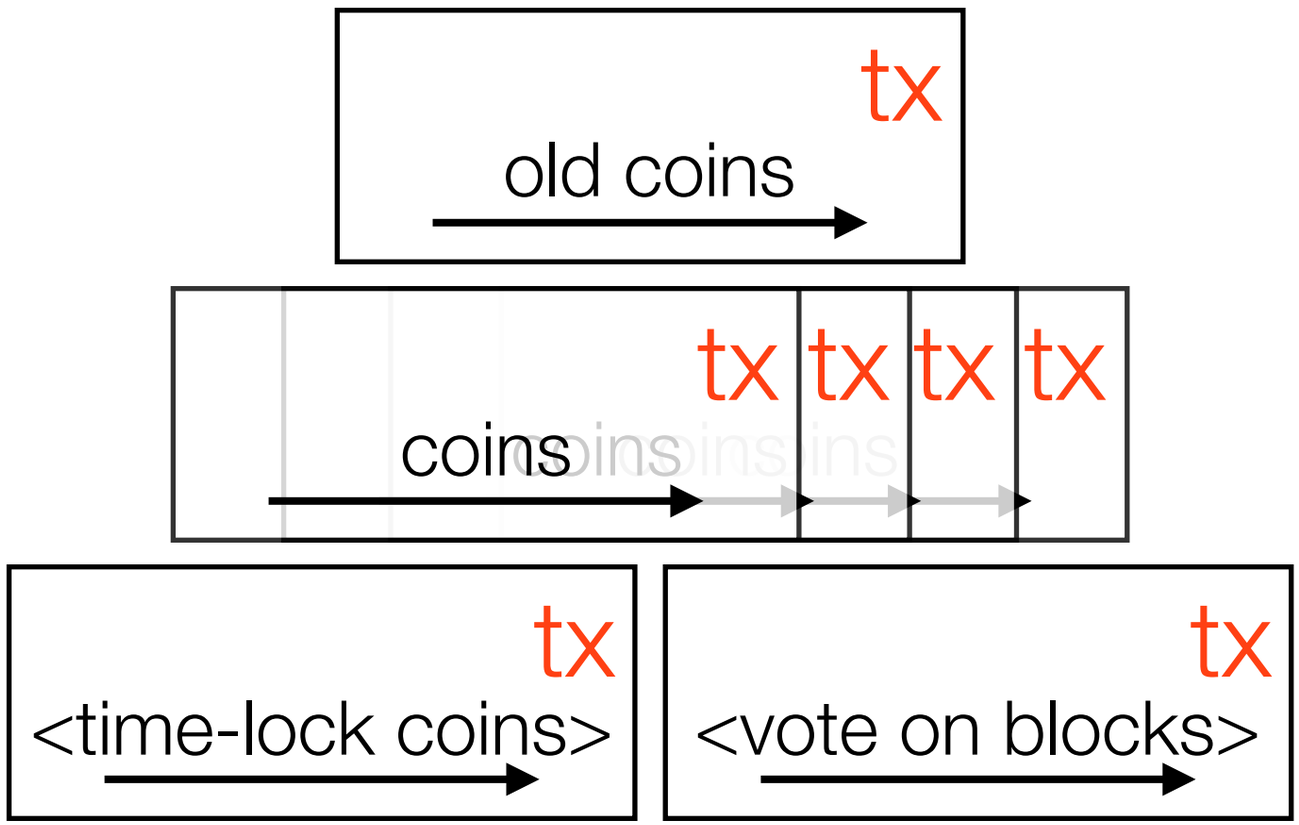
“proof of use”

(icon by lastspark for the noun project)

proof of stake

use of computational resources represents stake in system

other forms of stake?



“proof of burn”
“proof of coin age”

“proof of use”

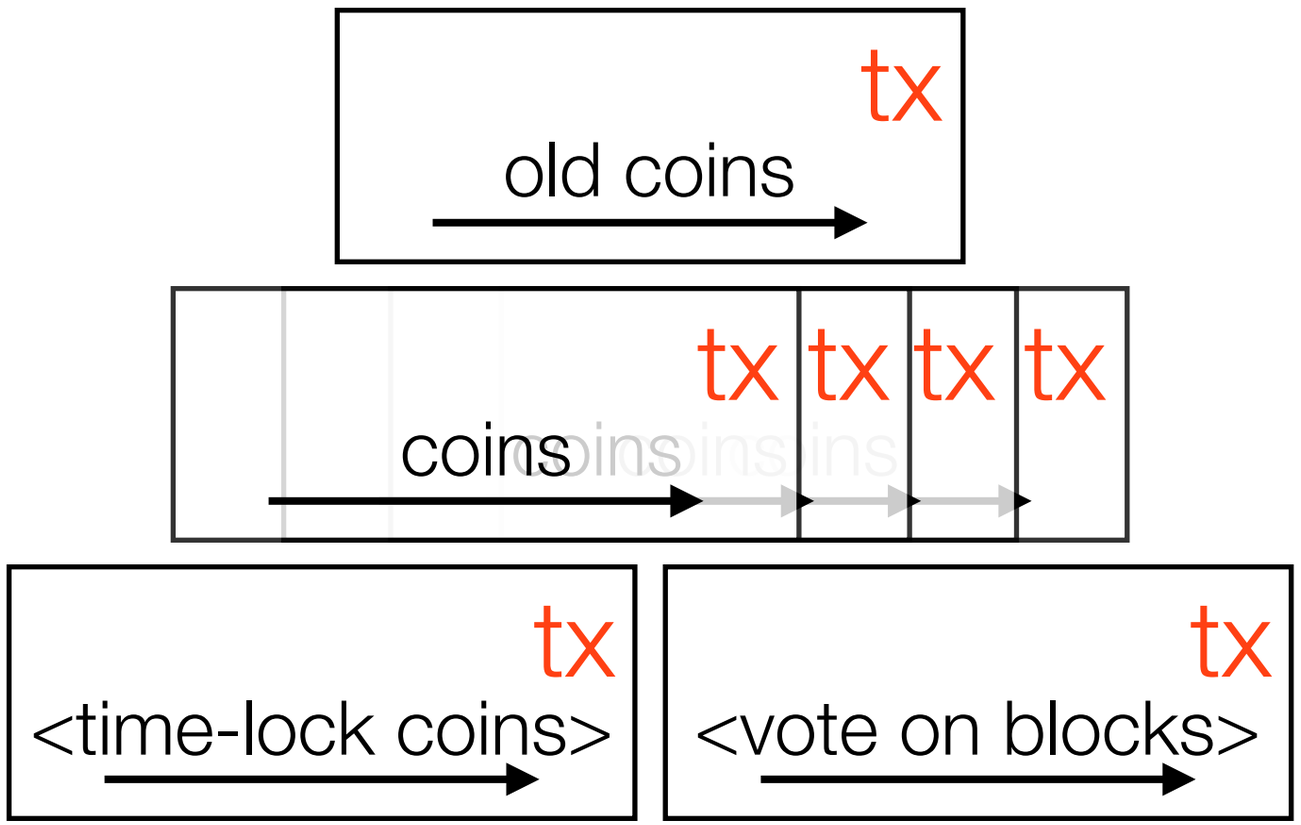
“security-deposit PoS”

(icon by lastspark for the noun project)

proof of stake

use of computational resources represents stake in system

other forms of stake?



“proof of burn”
“proof of coin age”

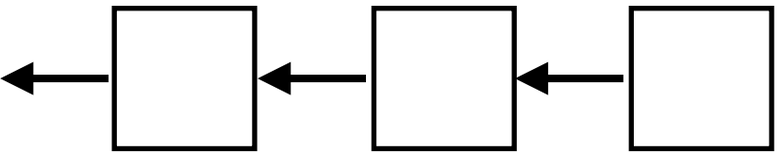
“proof of use”

“security-deposit PoS”

are these secure? how can we tell?

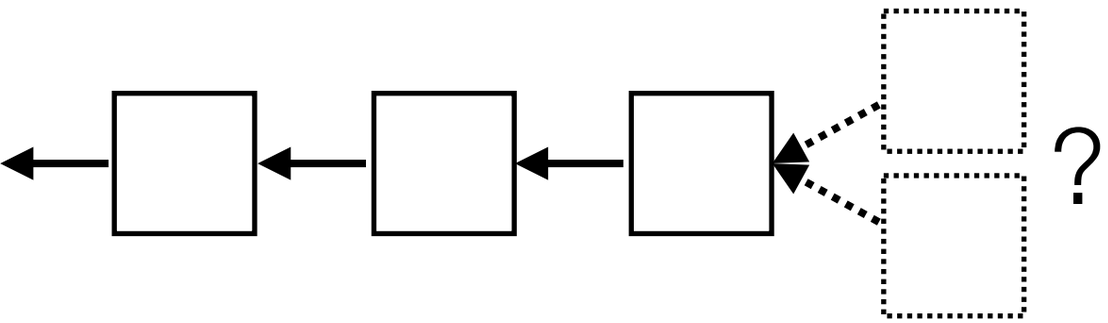
(icon by lastspark for the noun project)

proof of stake 🏆



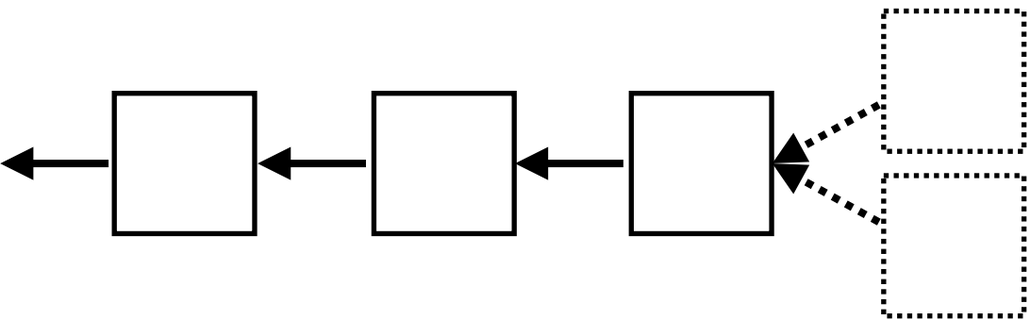
(icon by lastspark for the noun project)

proof of stake 



(icon by lastspark for the noun project)

proof of stake

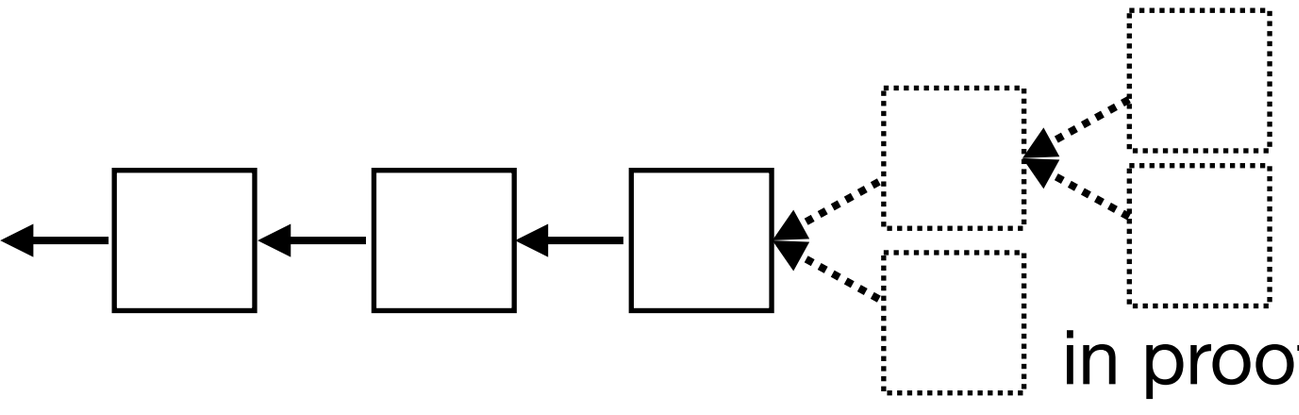


?

in proof-of-work, **cost** means you choose

(icon by lastspark for the noun project)

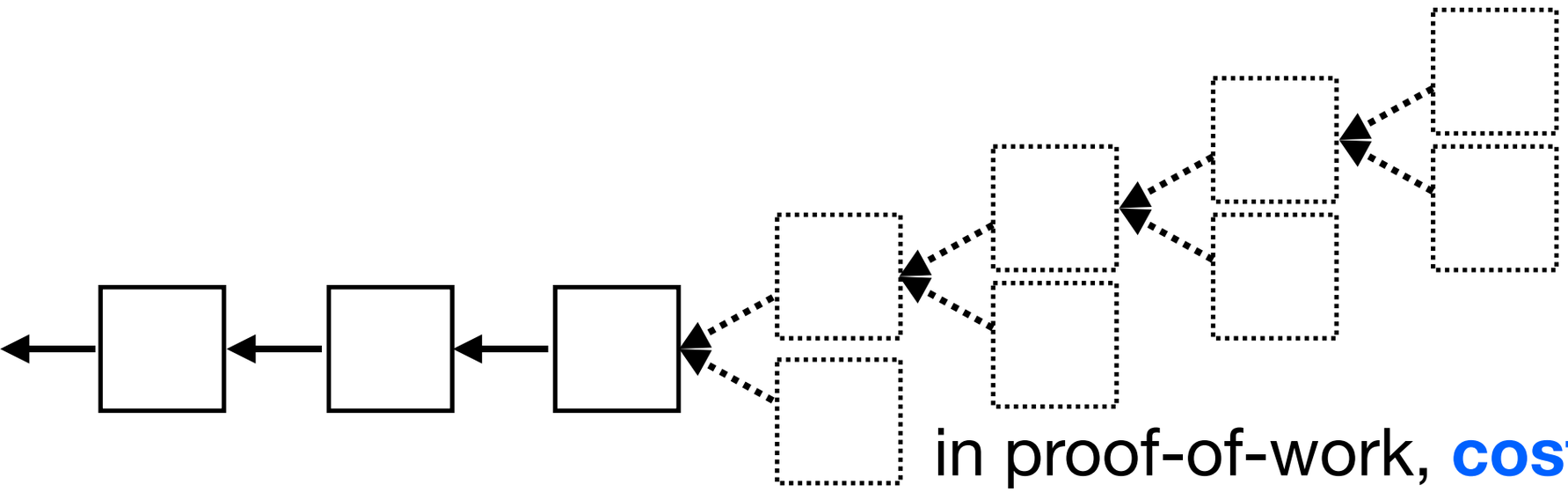
proof of stake



in proof-of-work, **cost** means you choose

(icon by lastspark for the noun project)

proof of stake

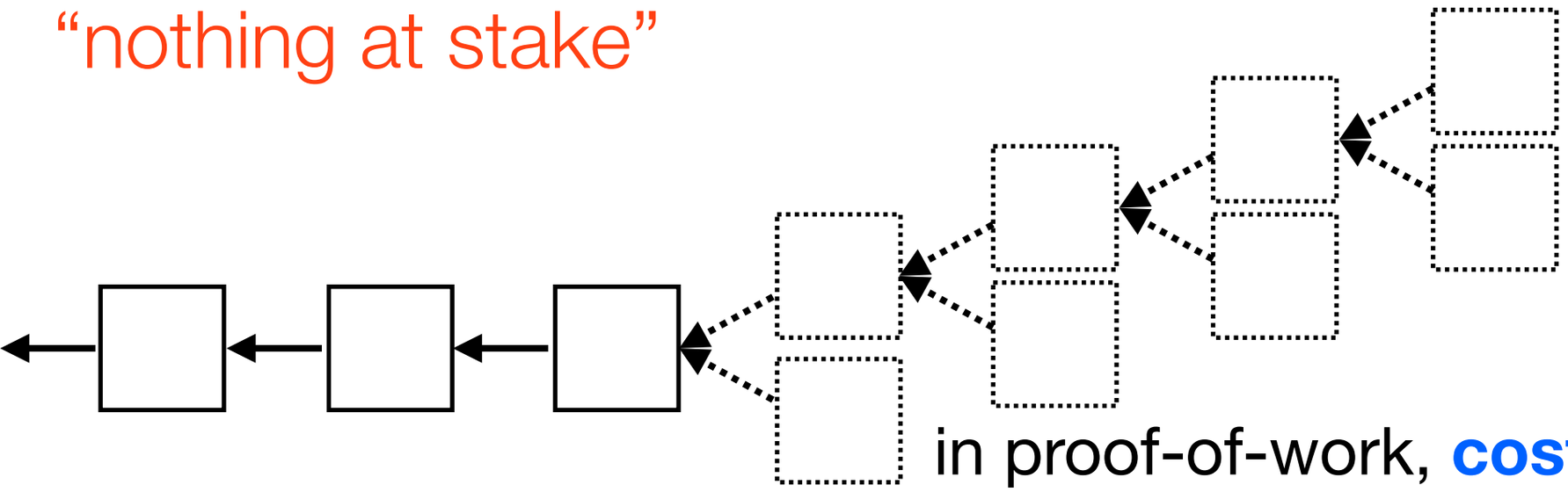


in proof-of-work, **cost** means you choose

(icon by lastspark for the noun project)

proof of stake

"nothing at stake"

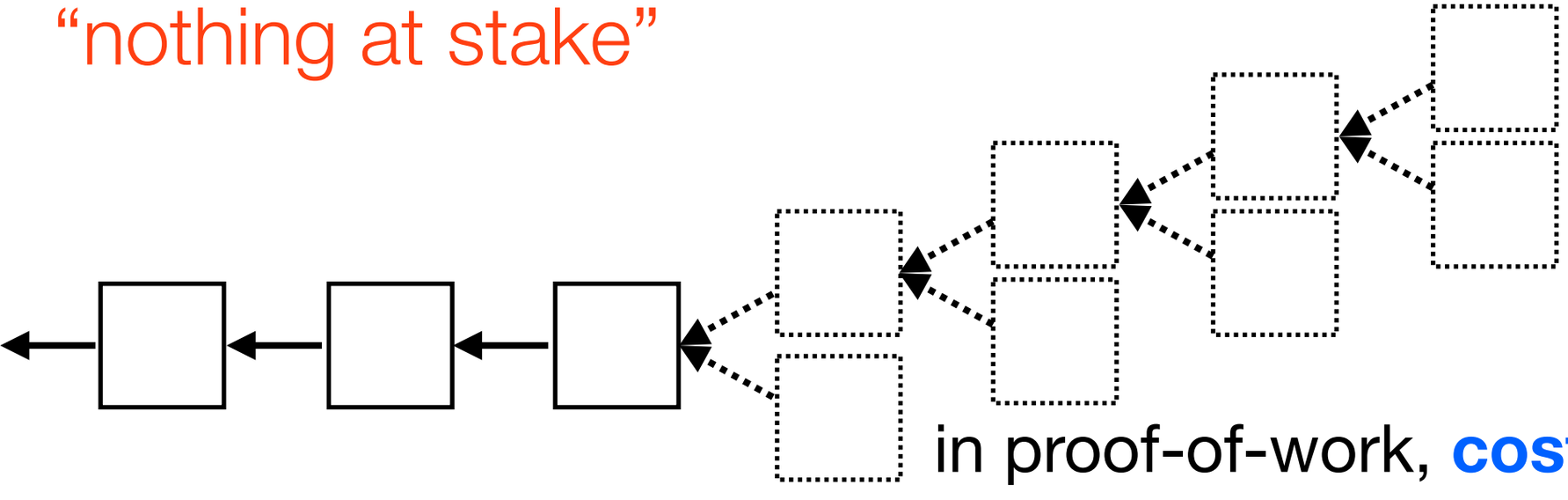


in proof-of-work, **cost** means you choose

(icon by lastspark for the noun project)

proof of stake

"nothing at stake"



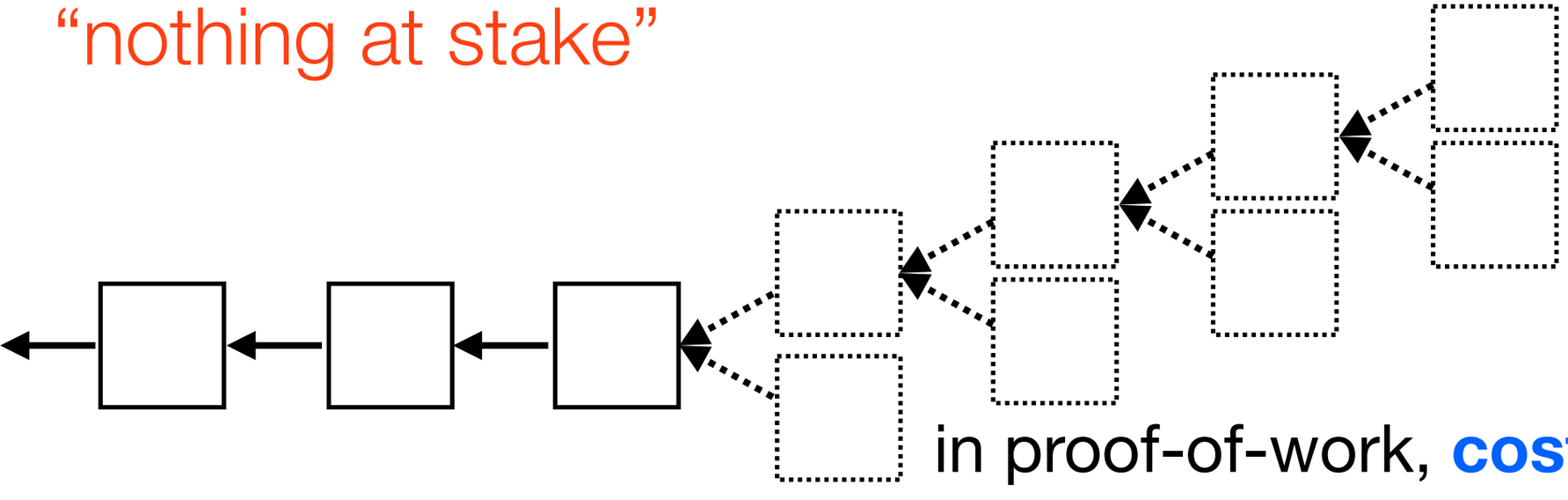
in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

(icon by lastspark for the noun project)

proof of stake

"nothing at stake"



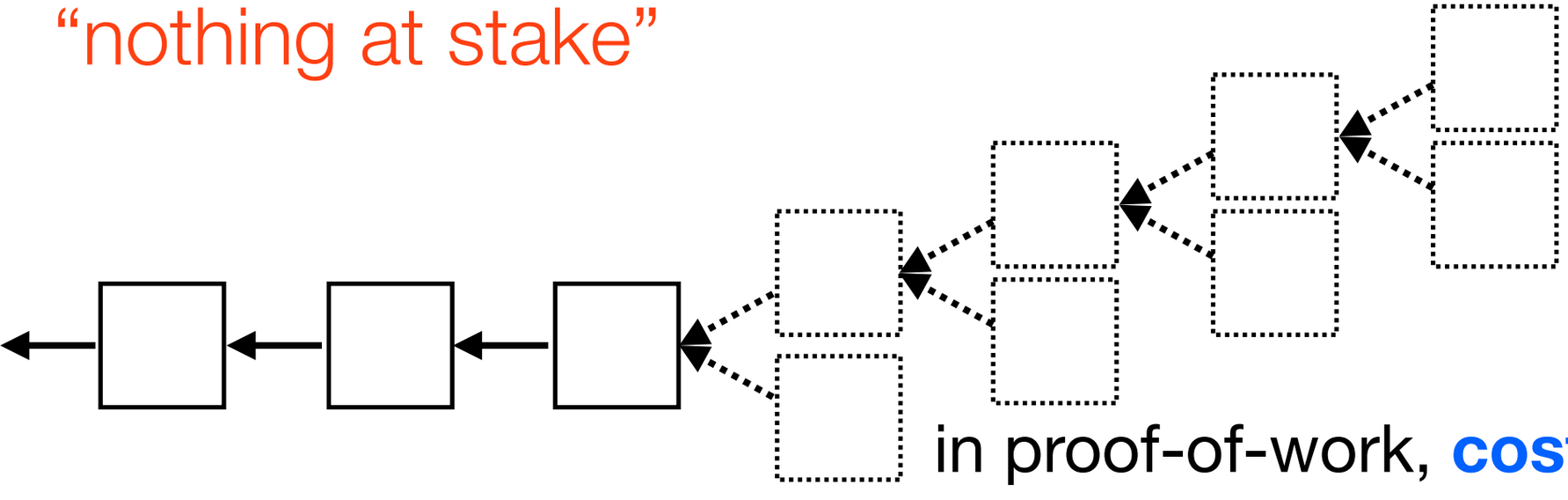
in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

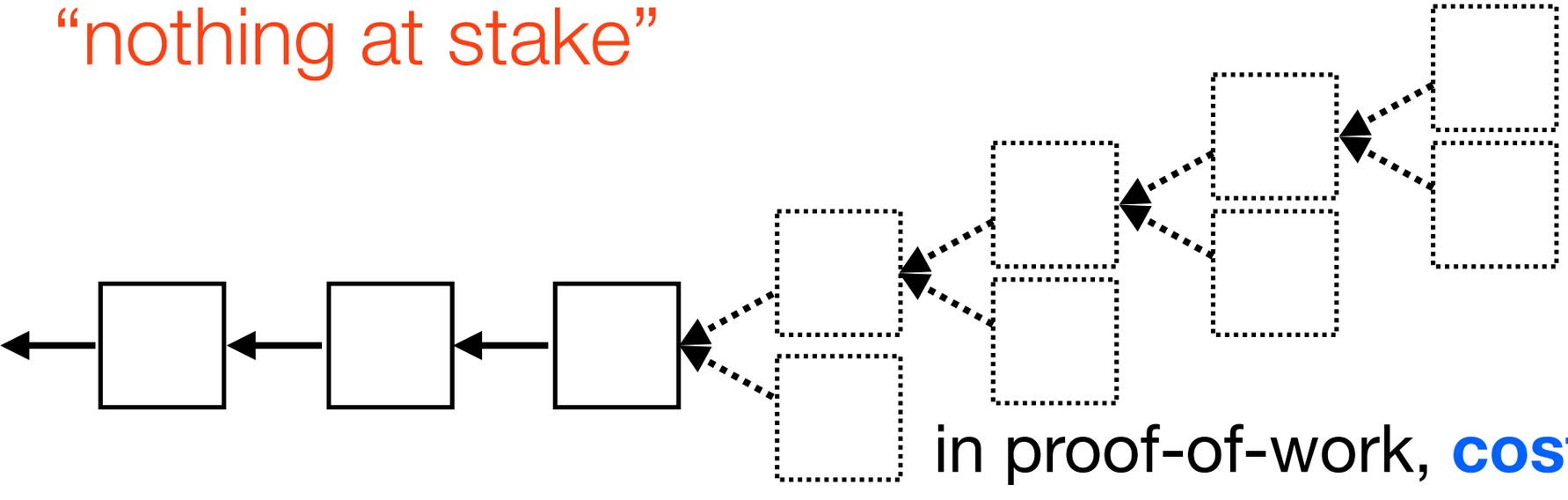
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

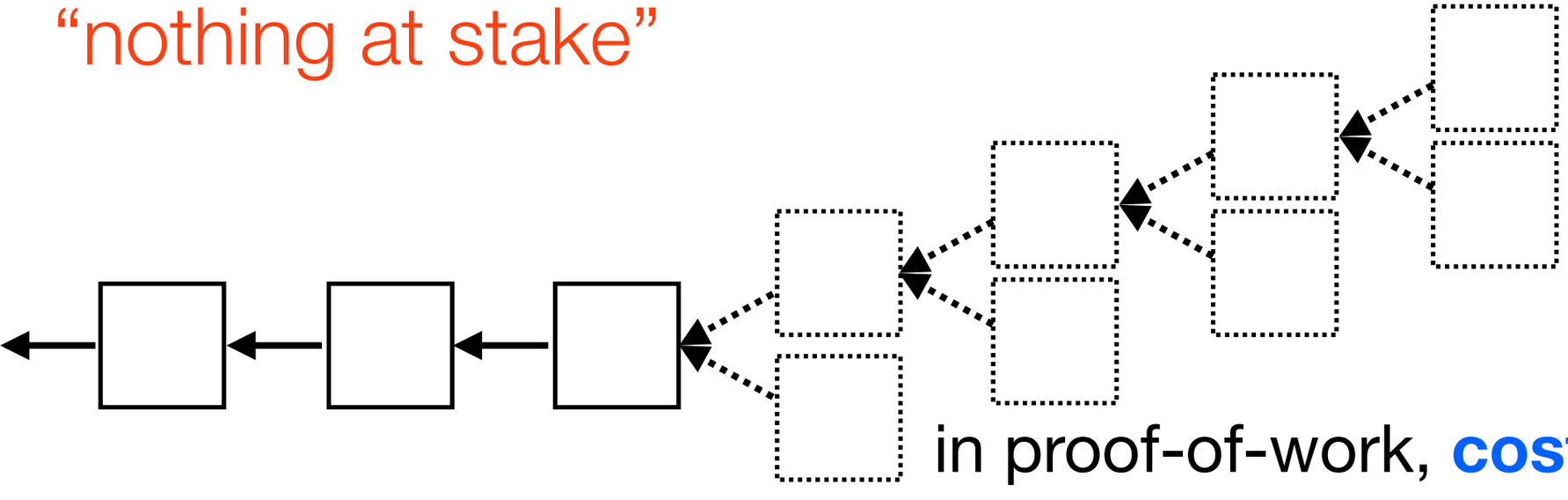
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

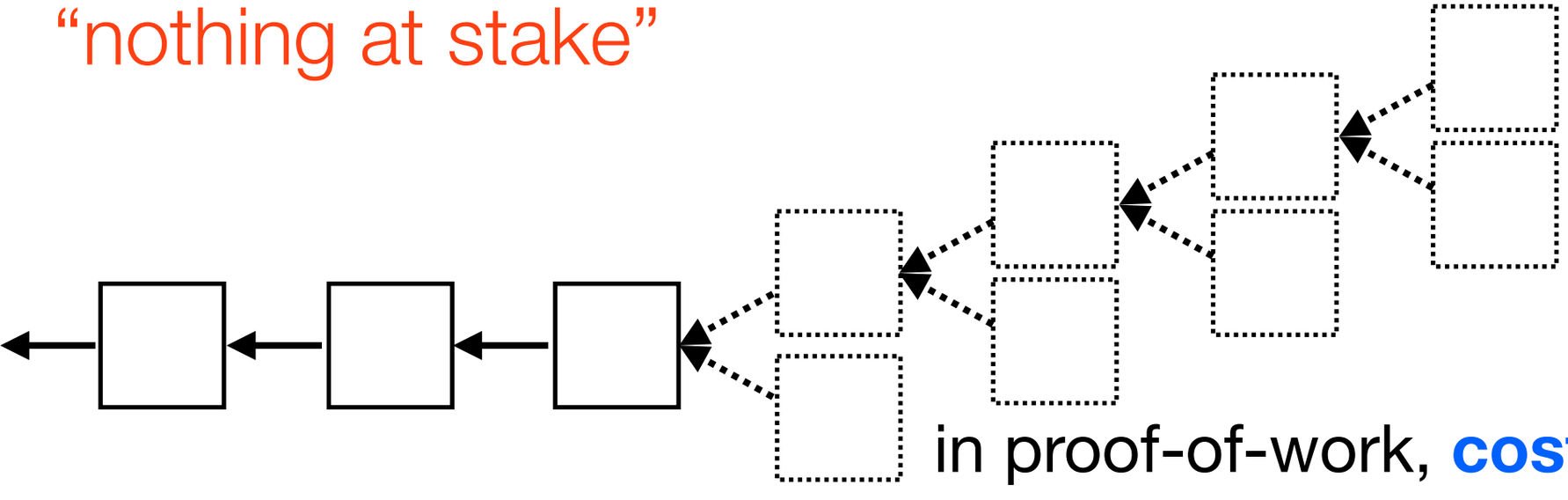
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

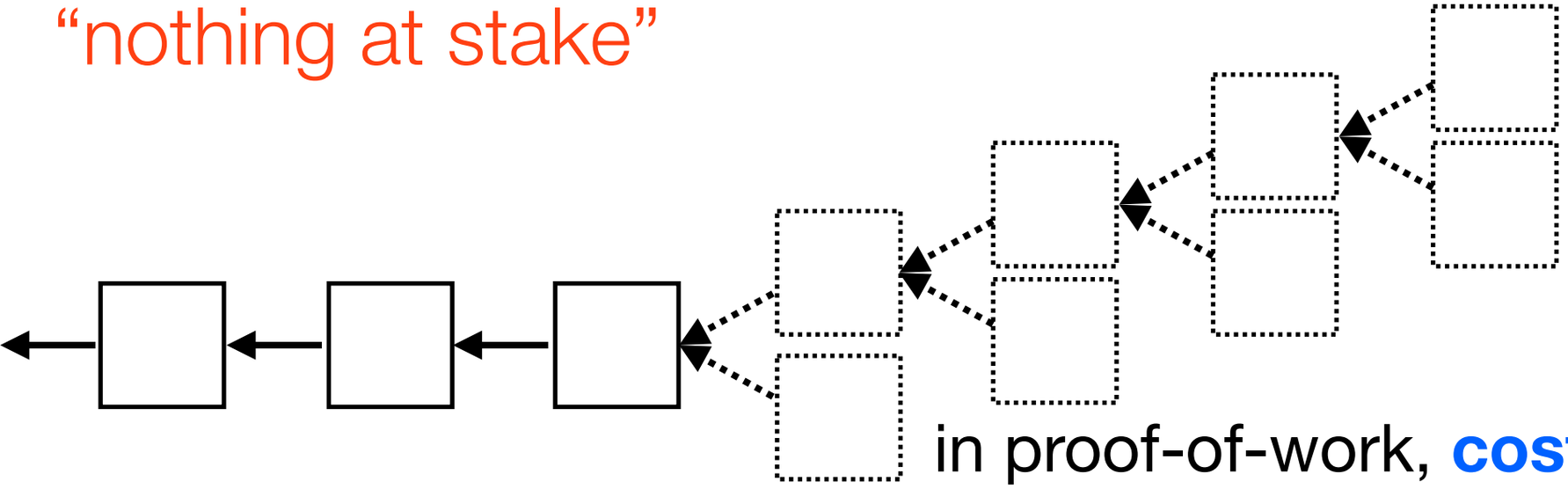
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

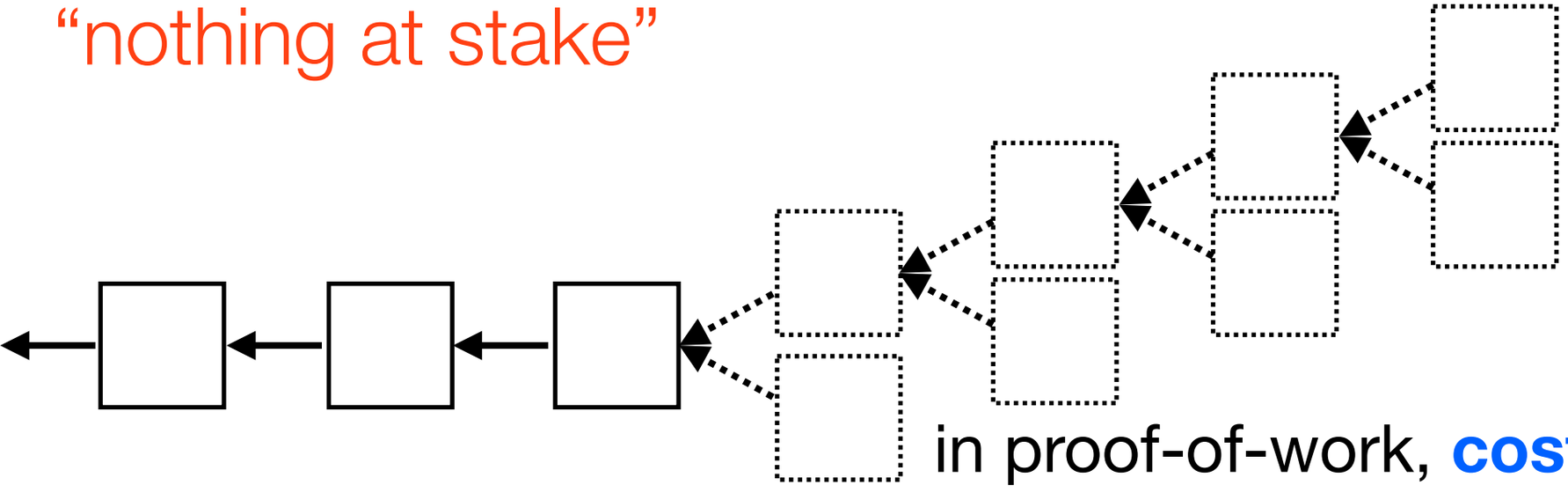
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

“stake grinding”

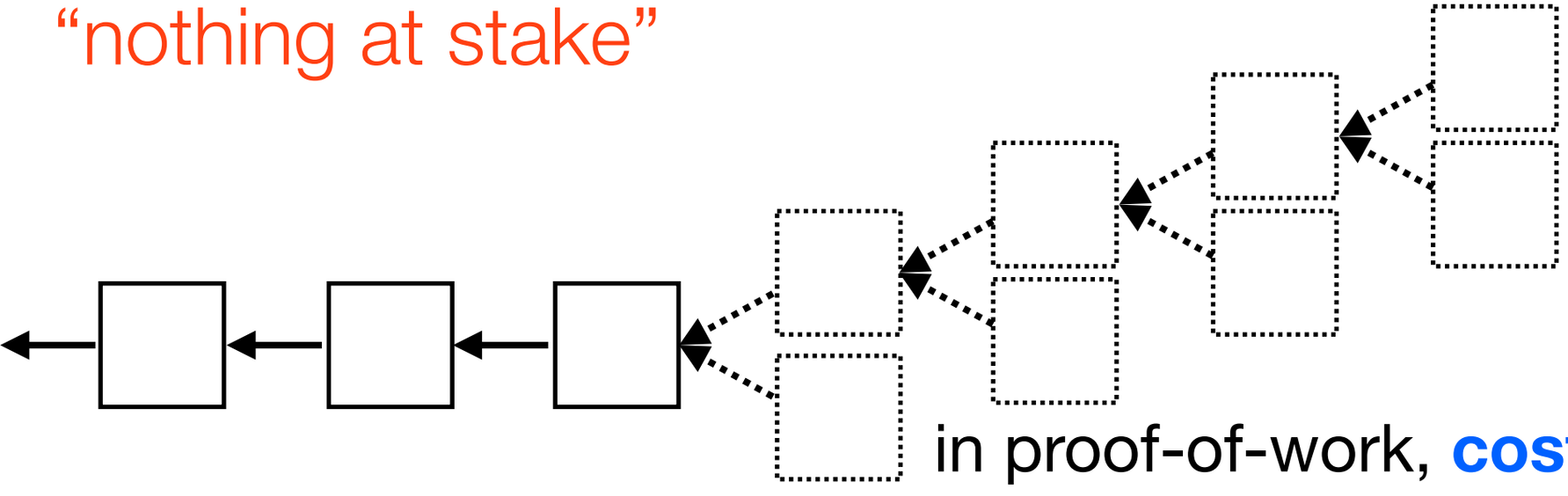
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

(icon by lastspark for the noun project)

proof of stake

“nothing at stake”



in proof-of-work, **cost** means you choose

in proof-of-stake, address with **punishment**

“stake grinding”

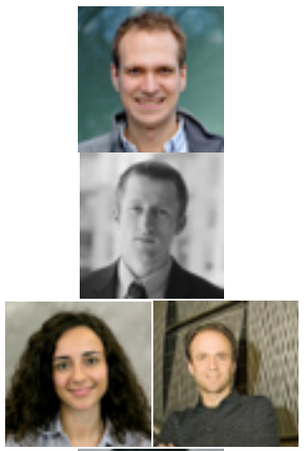
how do miners get chosen?

in proof-of-work, **can’t influence** this decision

in proof-of-stake, address with **stake modifiers**

(icon by lastspark for the noun project)

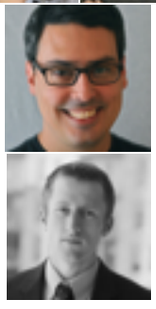
issues with Bitcoin



lack of fungibility

hashing rates are out of control

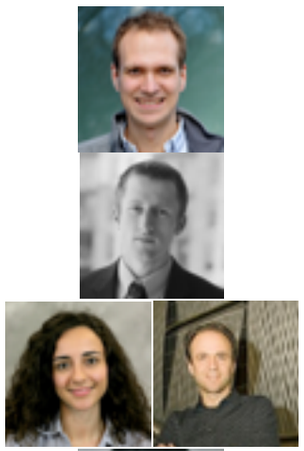
lack of scalability



lack of consumer protection

attacks on mining / misaligned incentives

issues with Bitcoin

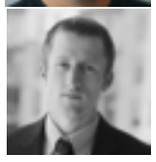


lack of fungibility

hashing rates are out of control

lack of scalability

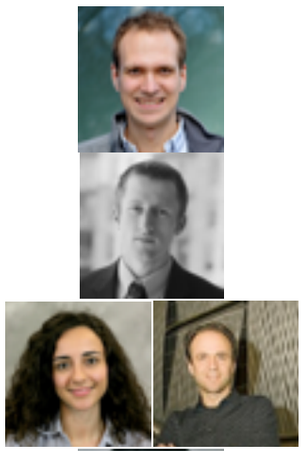
lack of consumer protection



attacks on mining / misaligned incentives

not suitable for many applications!

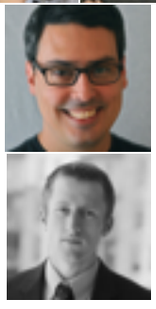
issues with Bitcoin



lack of fungibility

hashing rates are out of control

lack of scalability

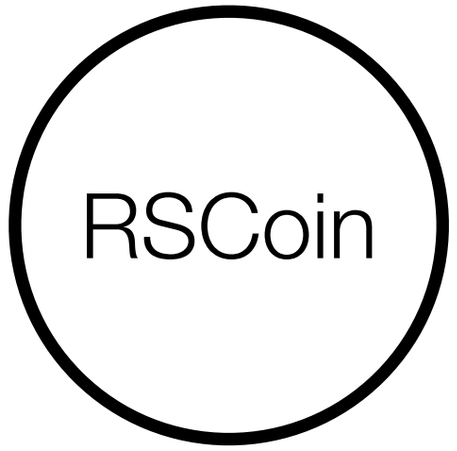


lack of consumer protection

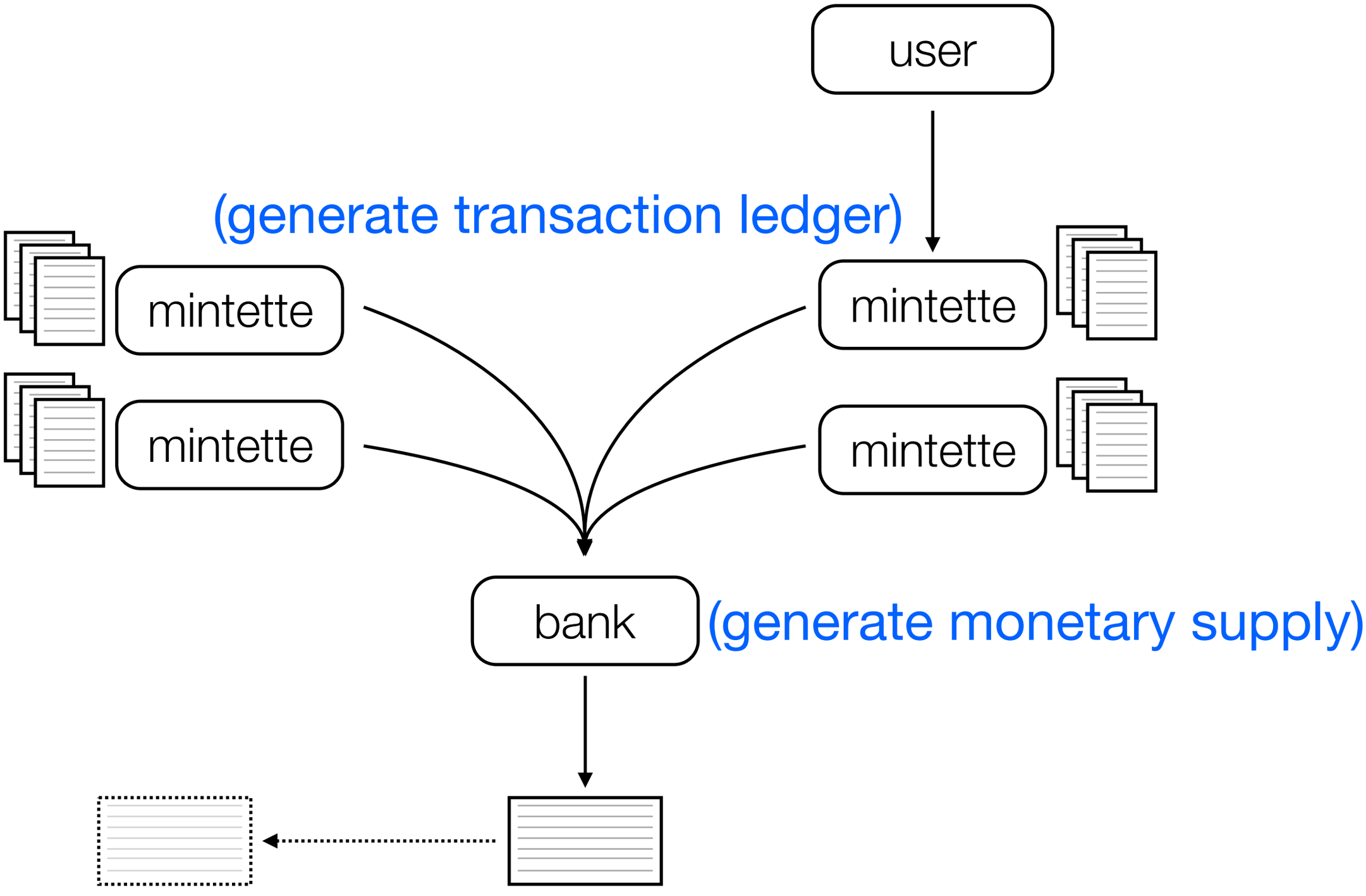
attacks on mining / misaligned incentives

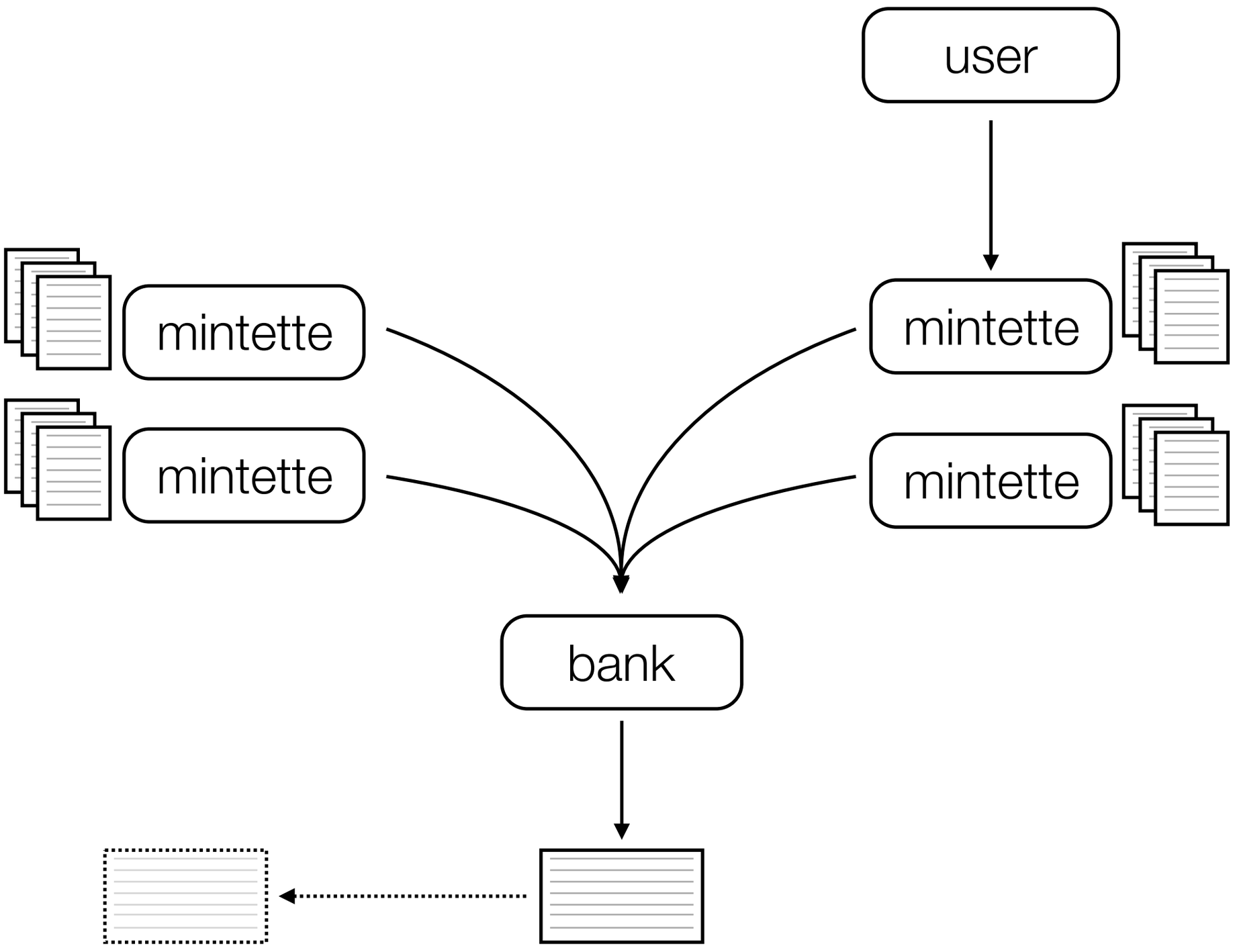
not suitable for many applications!

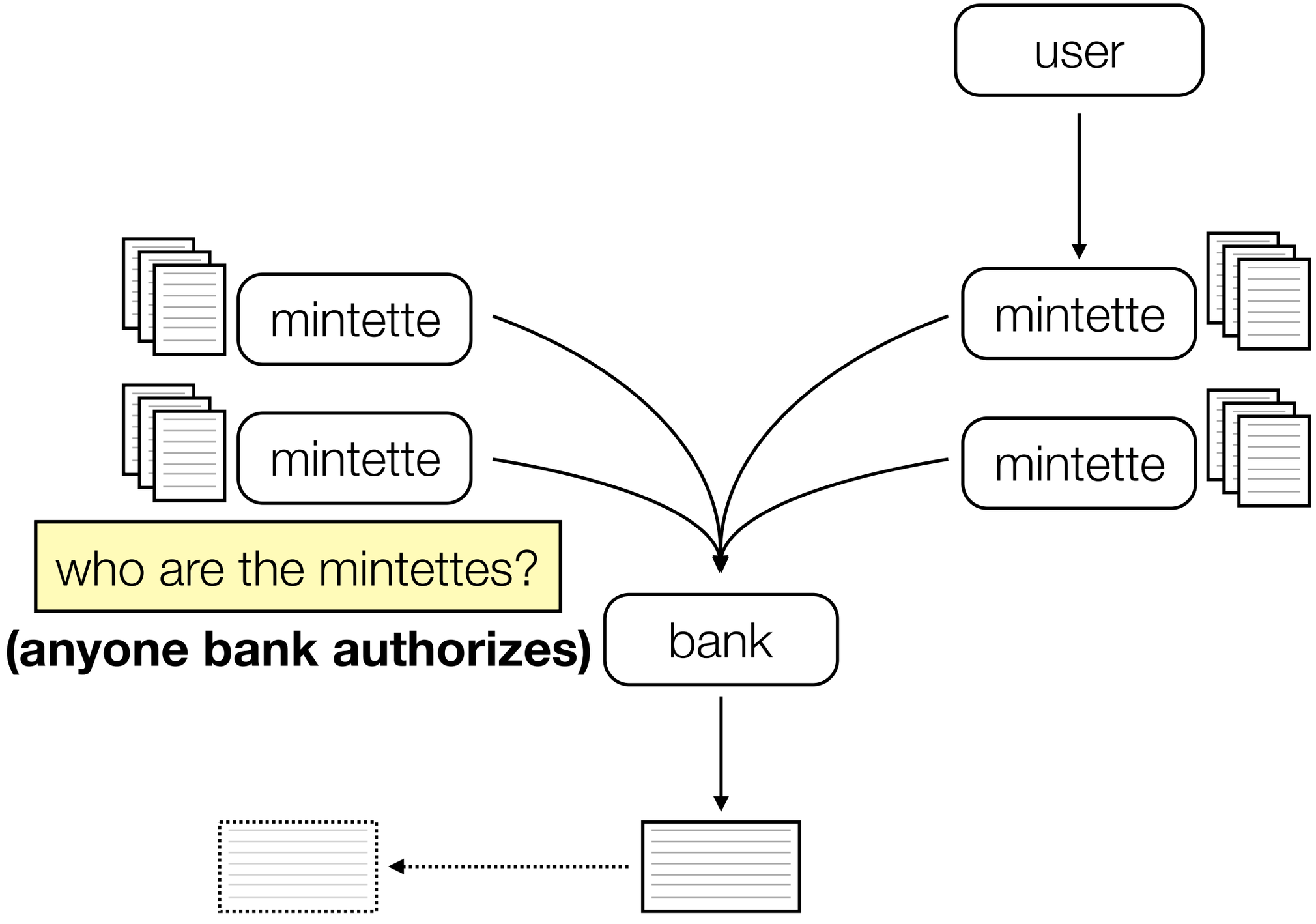
RSCoin [D**M** NDSS'16]

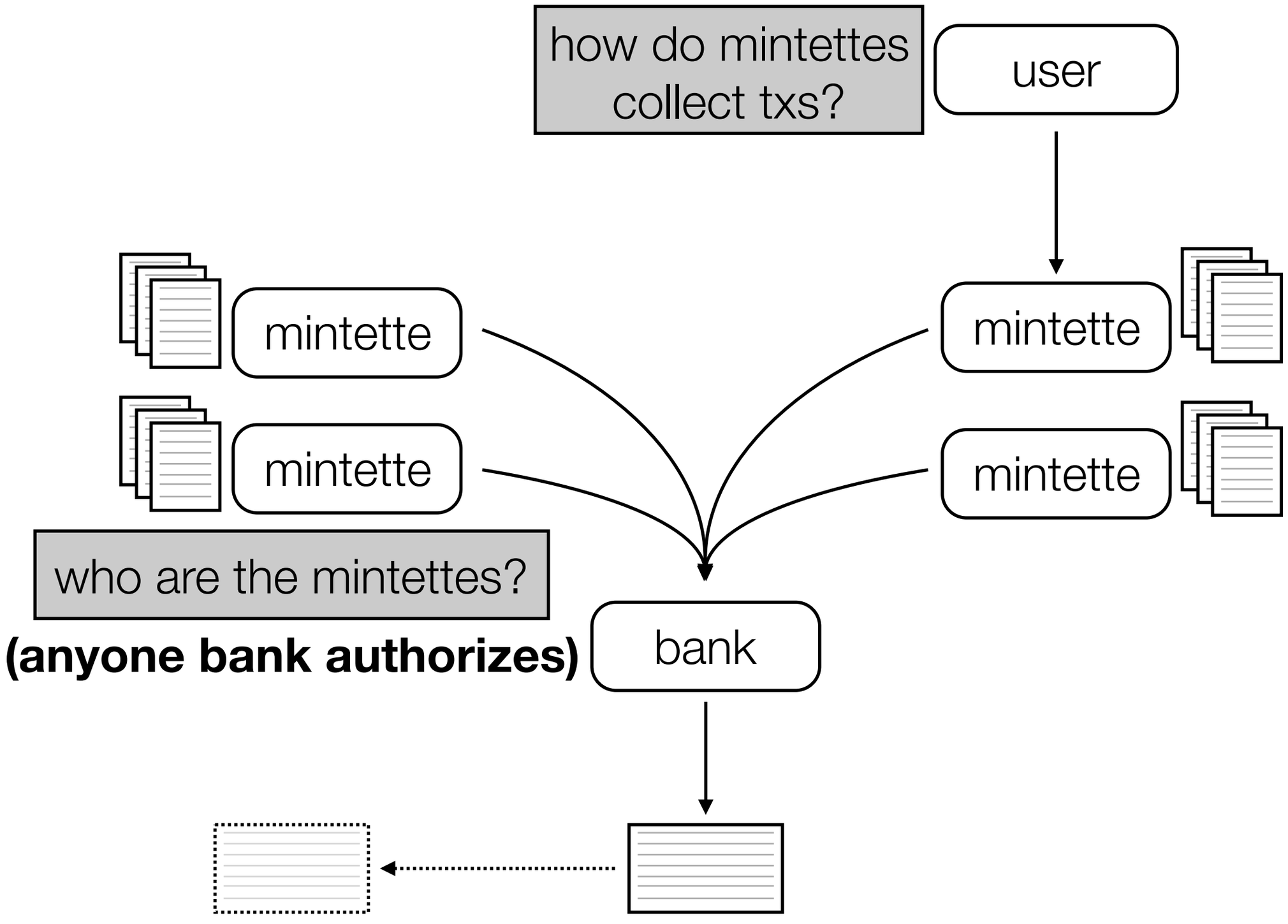


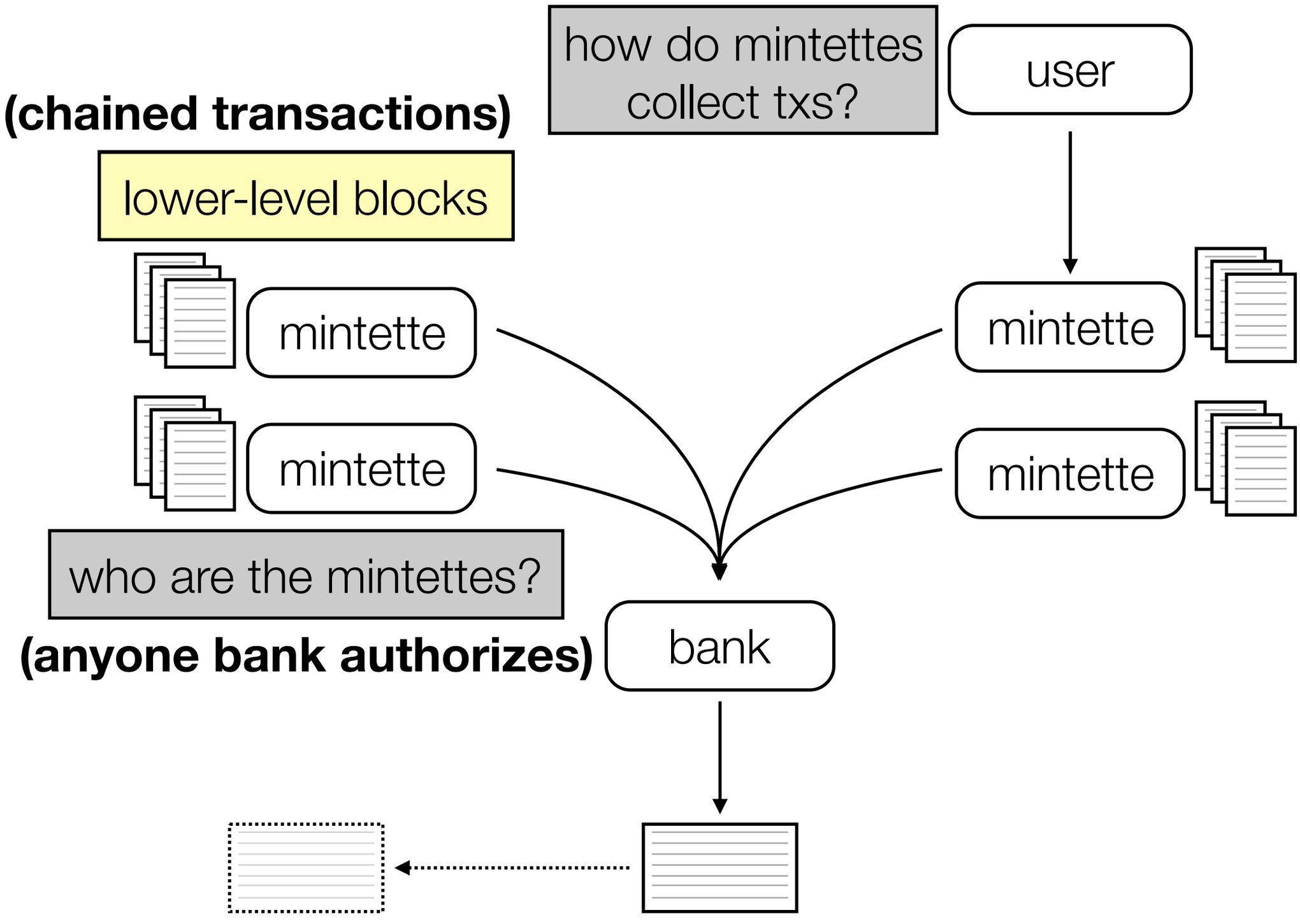
monetary supply	decentral	central	central
ledger	decentral	distribute	central
transparent?	y	y	n
pseudonyms?	y	y	n
computation	high!	low	low

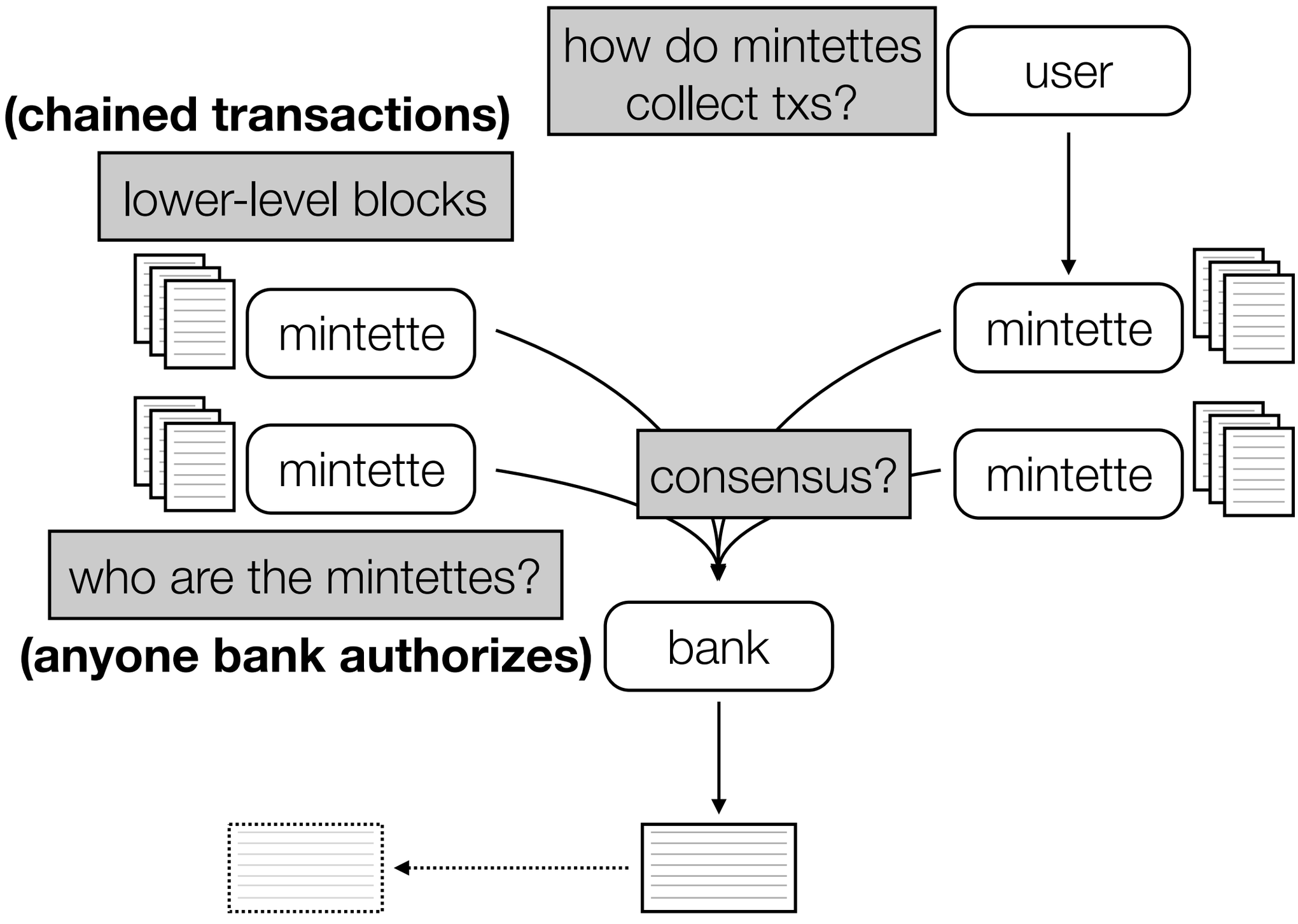

user
(generate transaction ledger)
mintette
mintette
mintette
mintette
bank
(generate monetary supply)

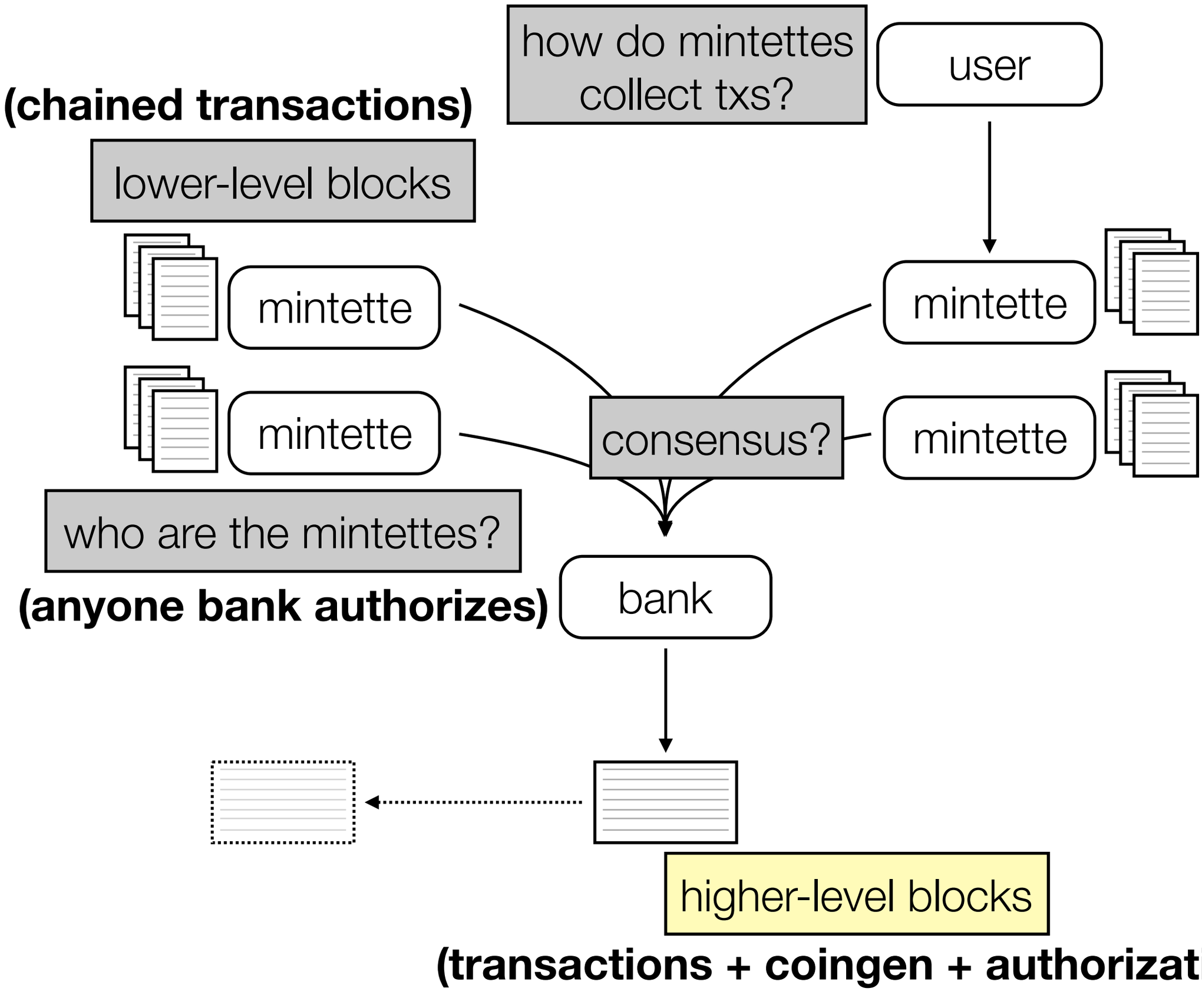

user
mintette
mintette
mintette
mintette
bank

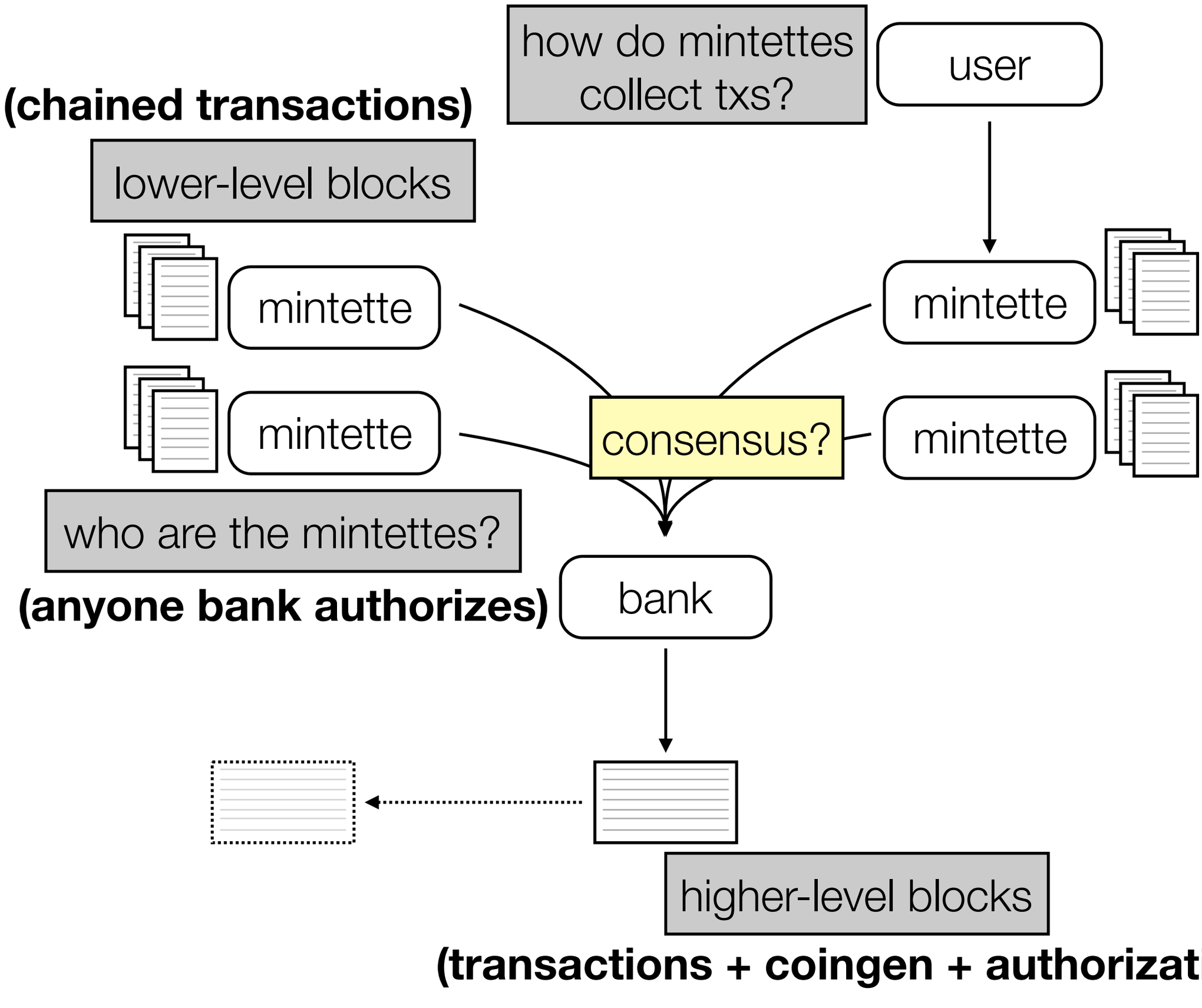

user
mintette
mintette
mintette
mintette
who are the mintettes?
(anyone bank authorizes)
bank


how do mintettes collect txs?
user
mintette
mintette
mintette
mintette
who are the mintettes?
(anyone bank authorizes)
bank

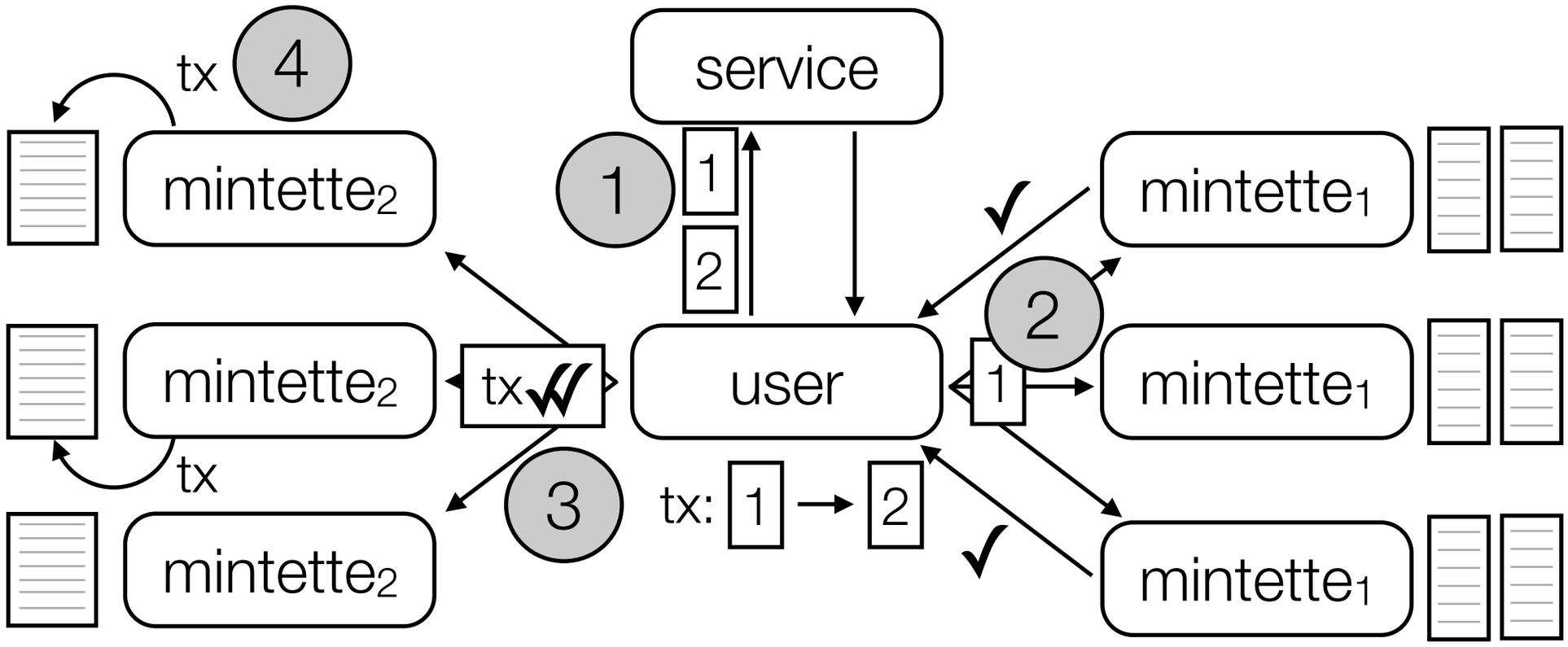

how do mintettes collect txs?
user
(chained transactions)
lower-level blocks
mintette
mintette
mintette
mintette
who are the mintettes?
(anyone bank authorizes)
bank


how do mintettes collect txs?
user
(chained transactions)
lower-level blocks
mintette
mintette
mintette
mintette
consensus?
who are the mintettes?
(anyone bank authorizes)
bank

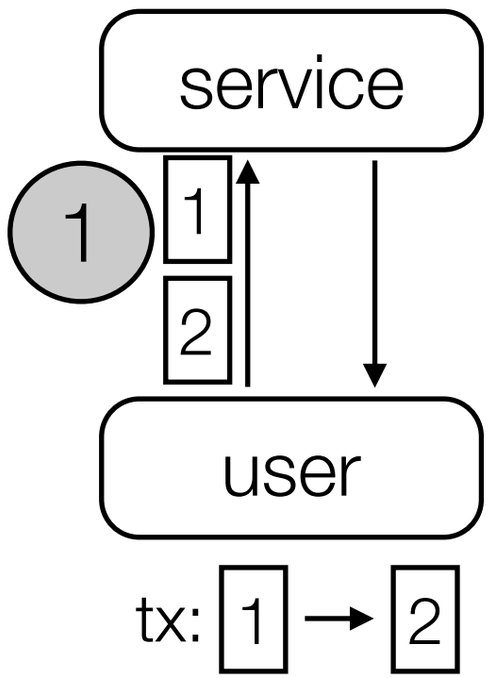

how do mintettes collect txs?
user
(chained transactions)
lower-level blocks
mintette
mintette
mintette
mintette
consensus?
who are the mintettes?
(anyone bank authorizes)
bank
higher-level blocks
(transactions + coingen + authorization)

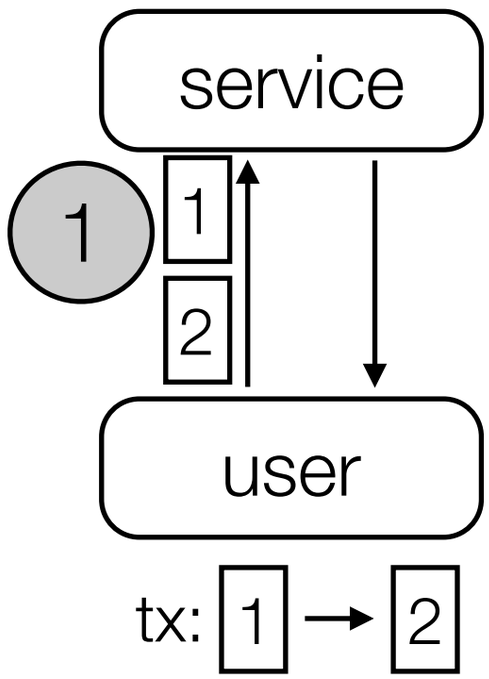

how do mintettes collect txs?
user
(chained transactions)
lower-level blocks
mintette
mintette
mintette
mintette
consensus?
who are the mintettes?
(anyone bank authorizes)
bank
higher-level blocks
(transactions + coingen + authorization)

consensus

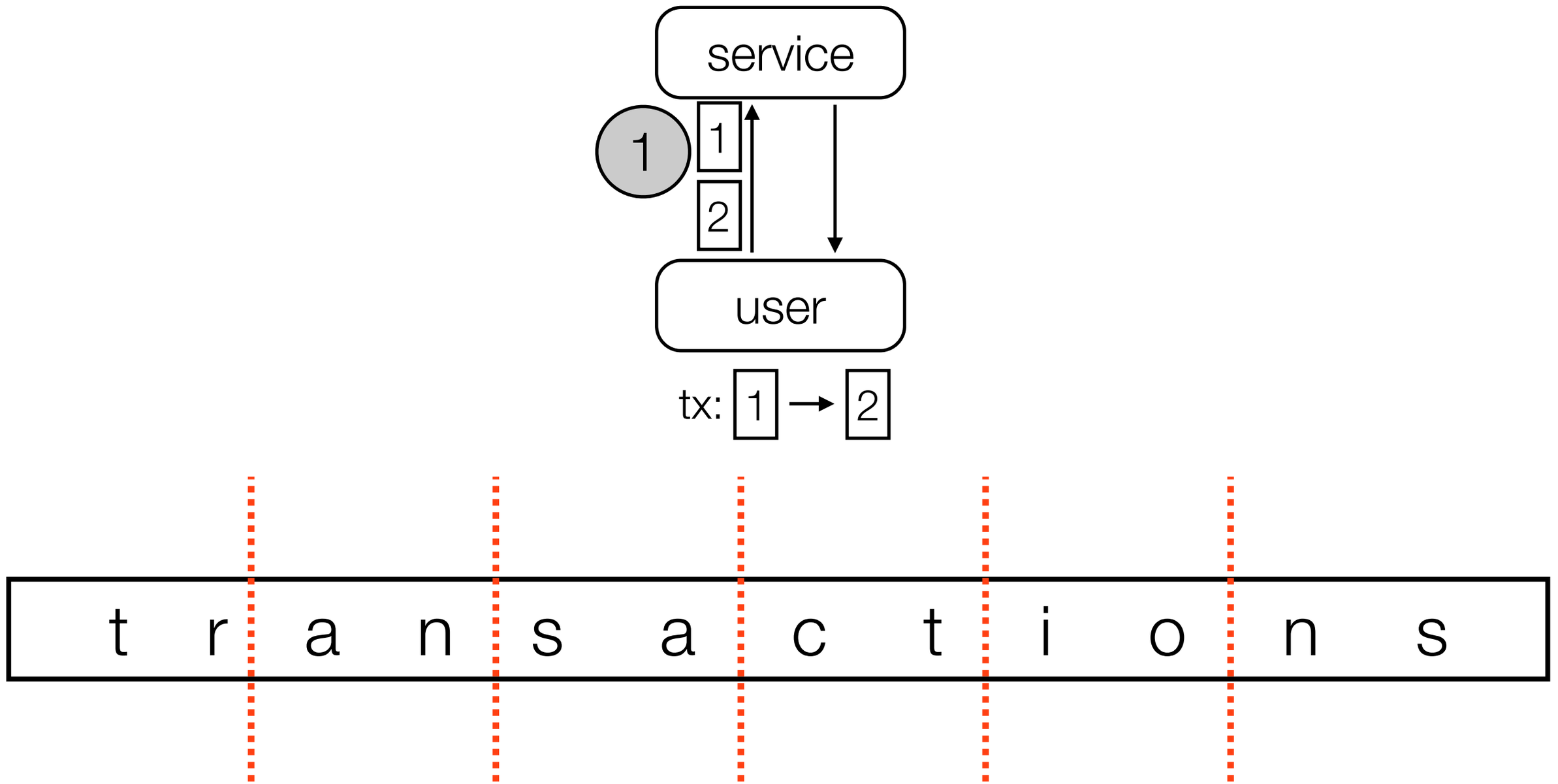


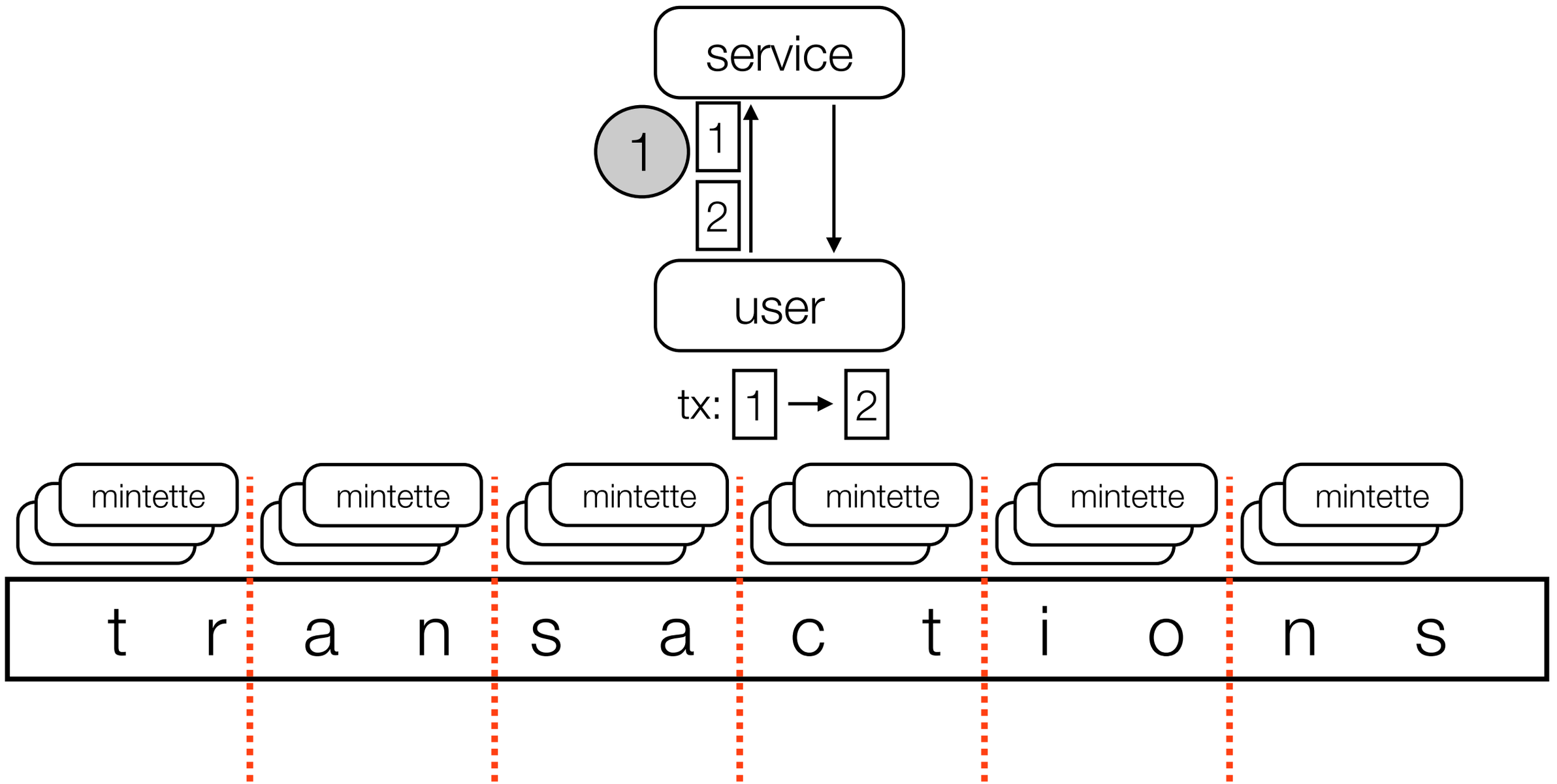
simple adaptation of Two-Phase Commit (2PC)

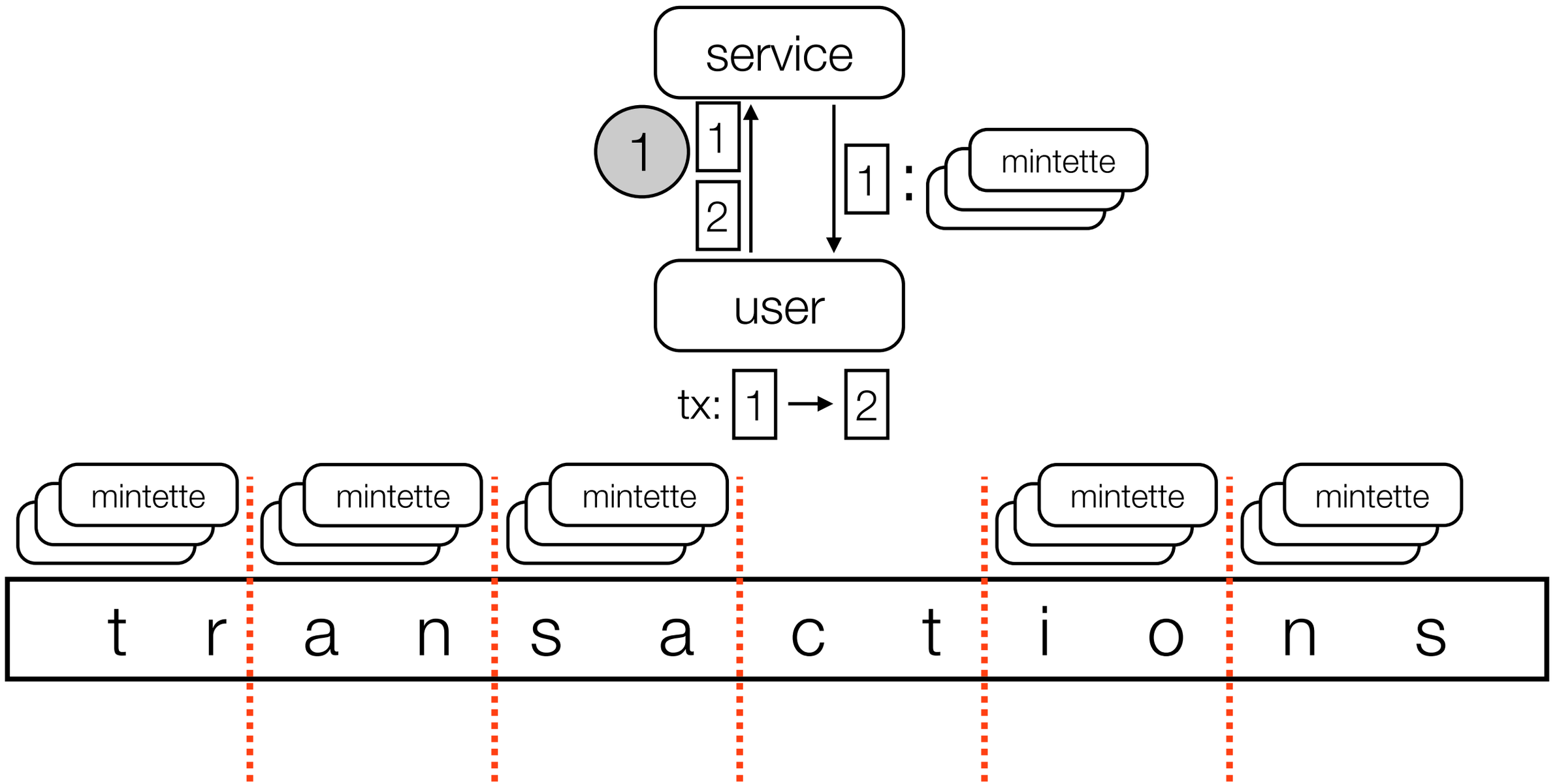

service
1
1
2
user
tx:
1
2

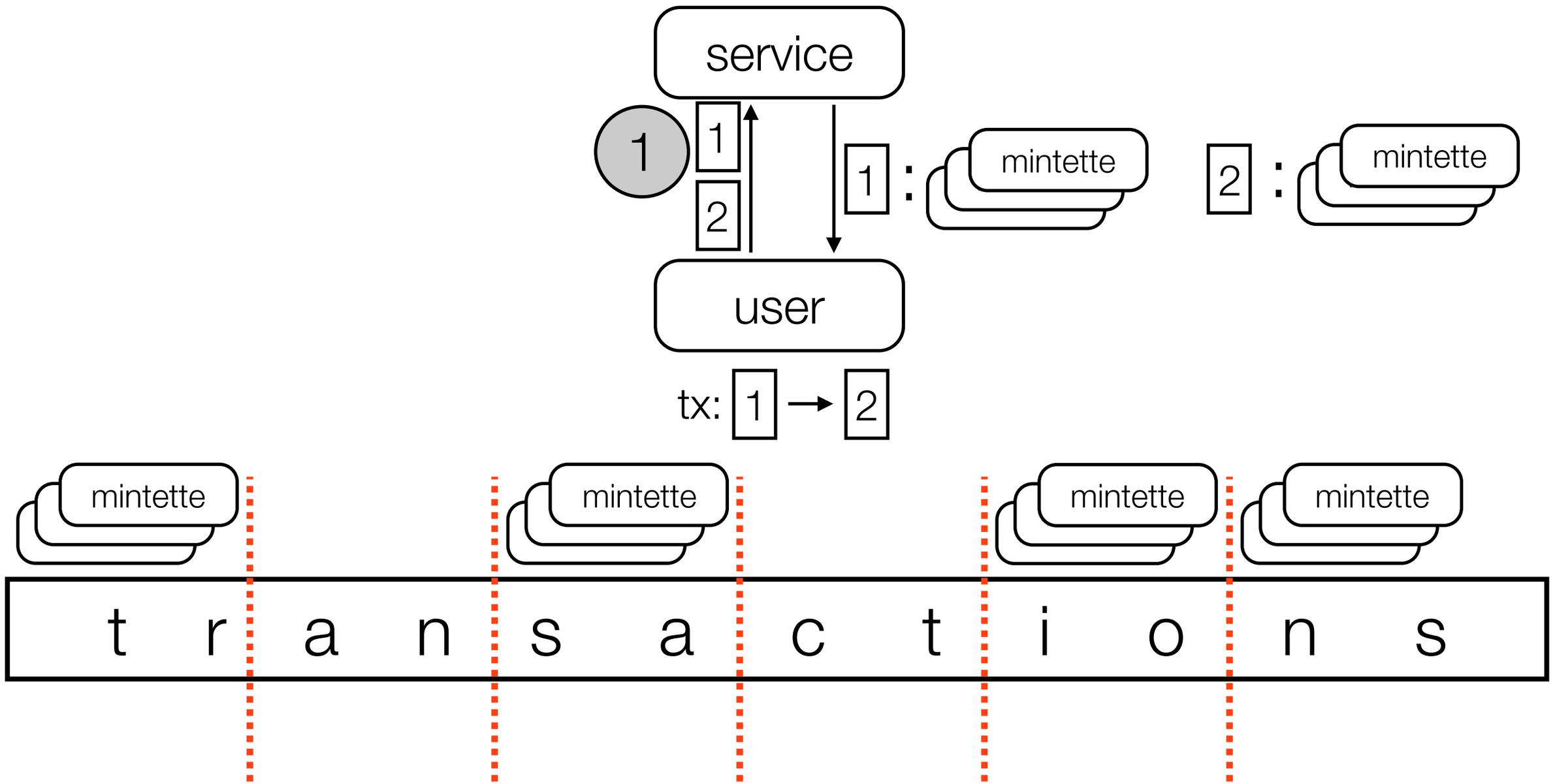


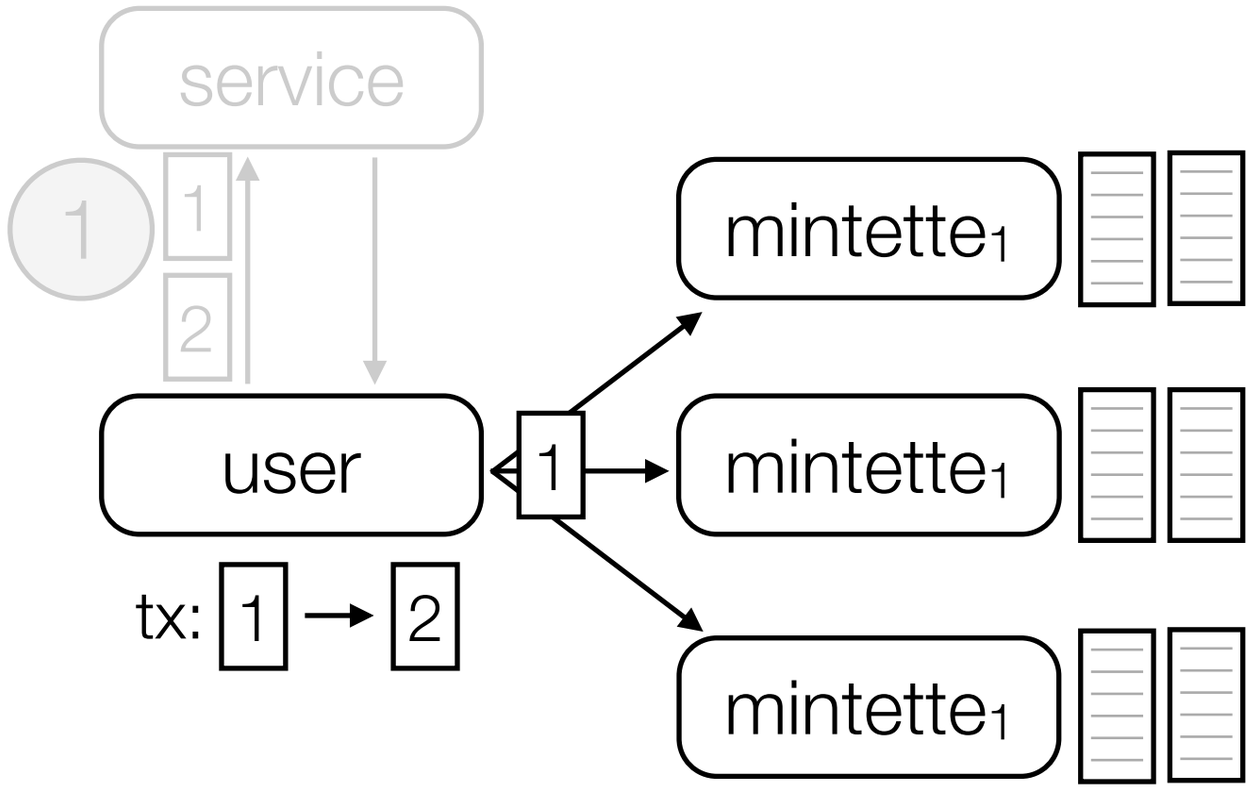
t r a n s a c t i o n s


service
1
1
2
user
tx: 1 → 2
t r a n s a c t i o n s

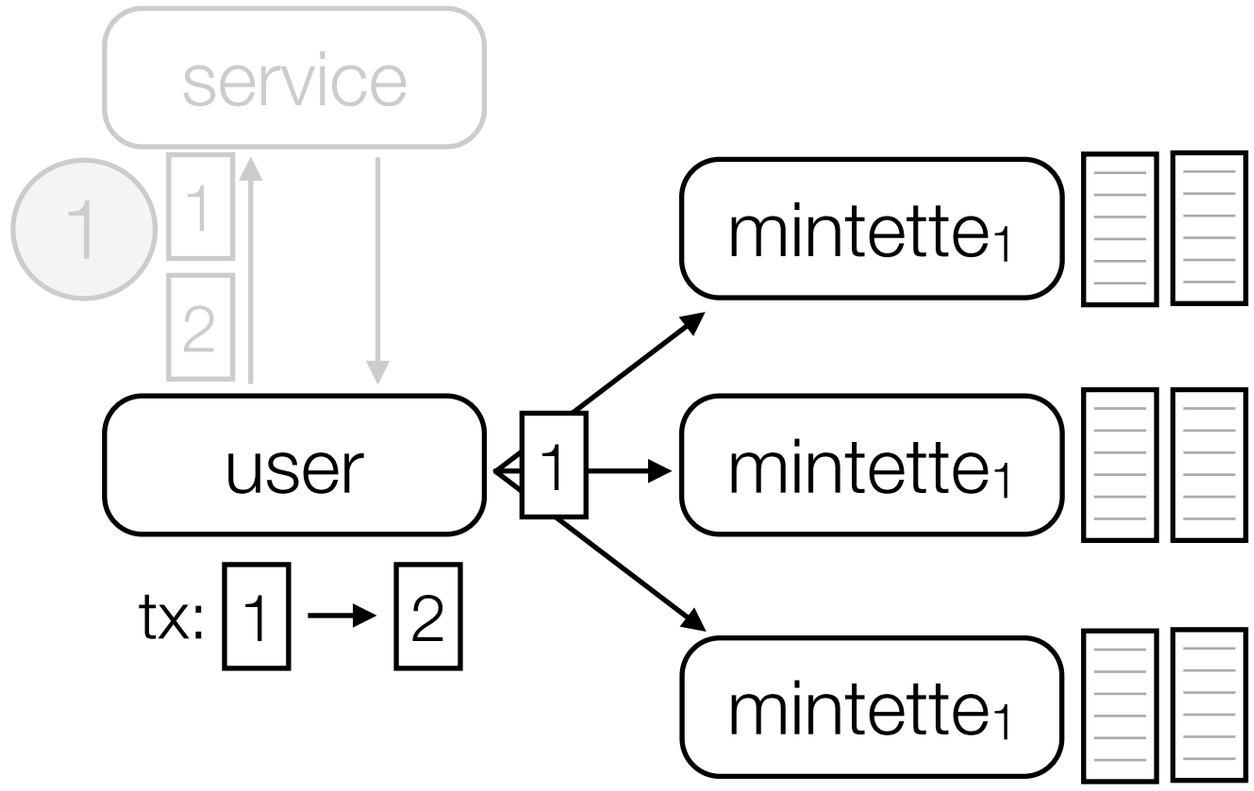

service
1
1
2
user
tx: 1 → 2
mintette
mintette
mintette
mintette
mintette
mintette
t r a n s a c t i o n s


service
1
1
2
1
mintette
user
tx: 1 → 2
mintette
mintette
mintette
mintette
mintette
transactions

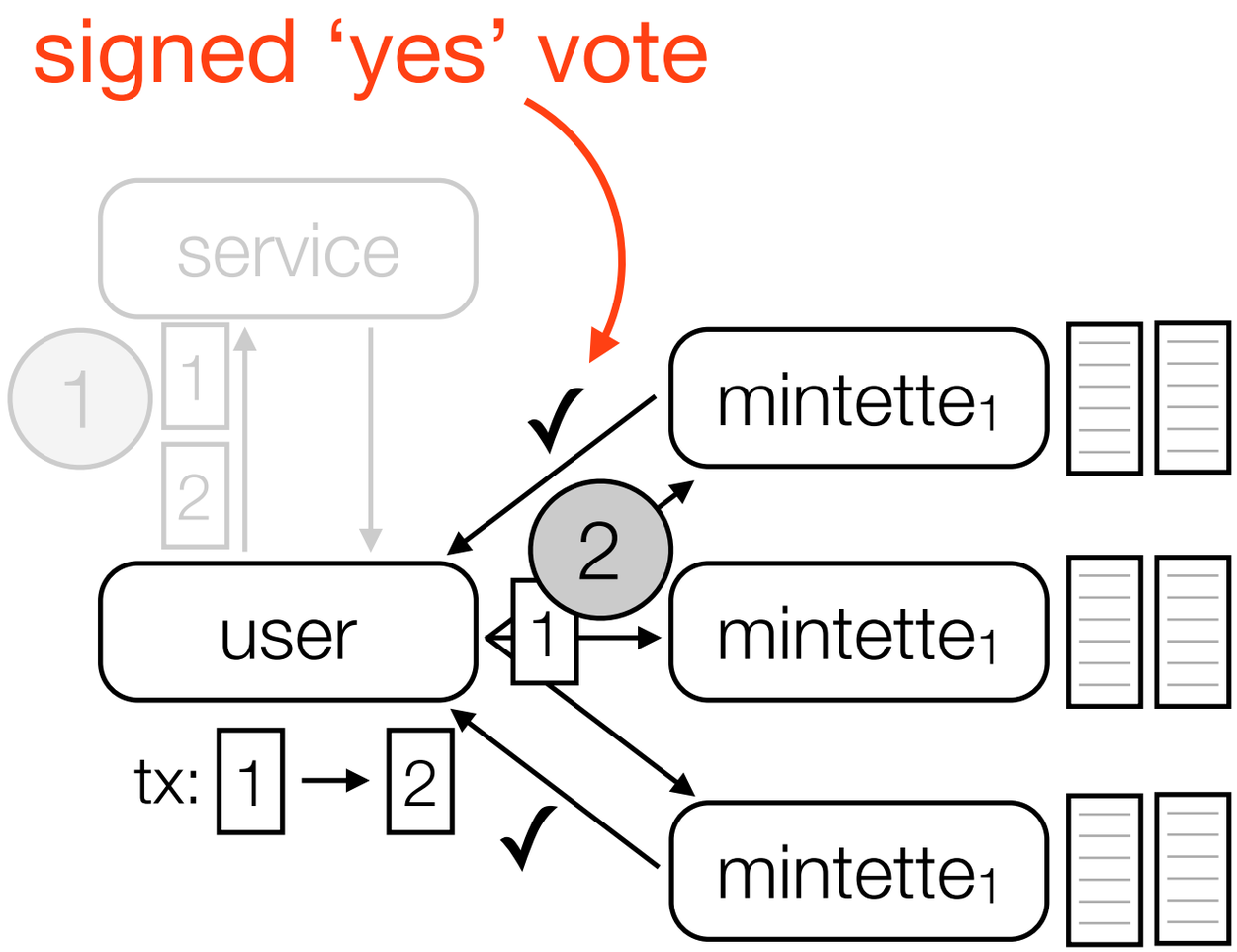

service
1
1
2
1 :
mintette
2 :
mintette
user
tx: 1 → 2
mintette
mintette
mintette
mintette
transactions

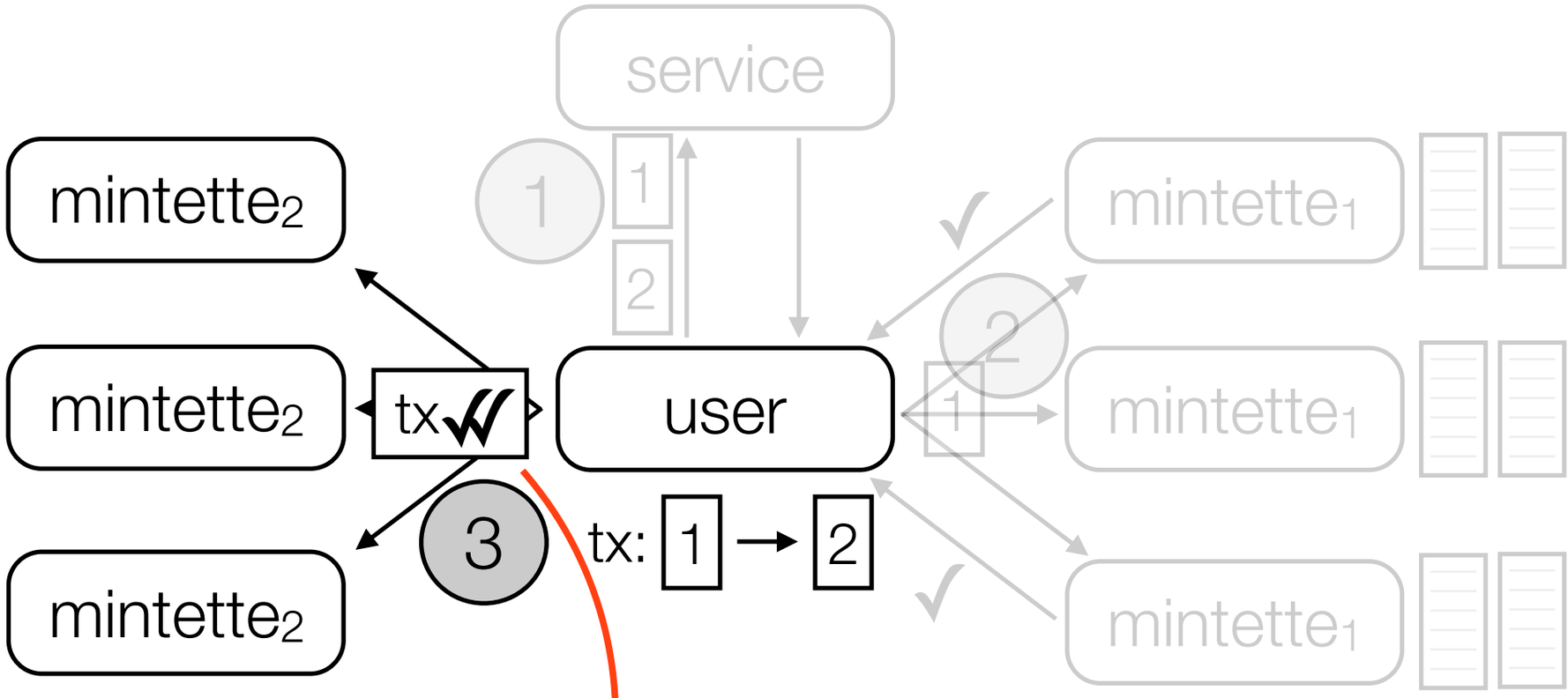

service
1
1
2
user
1
tx:
1
2
mintette1
mintette1
mintette1

mintettes check for **double spending**…



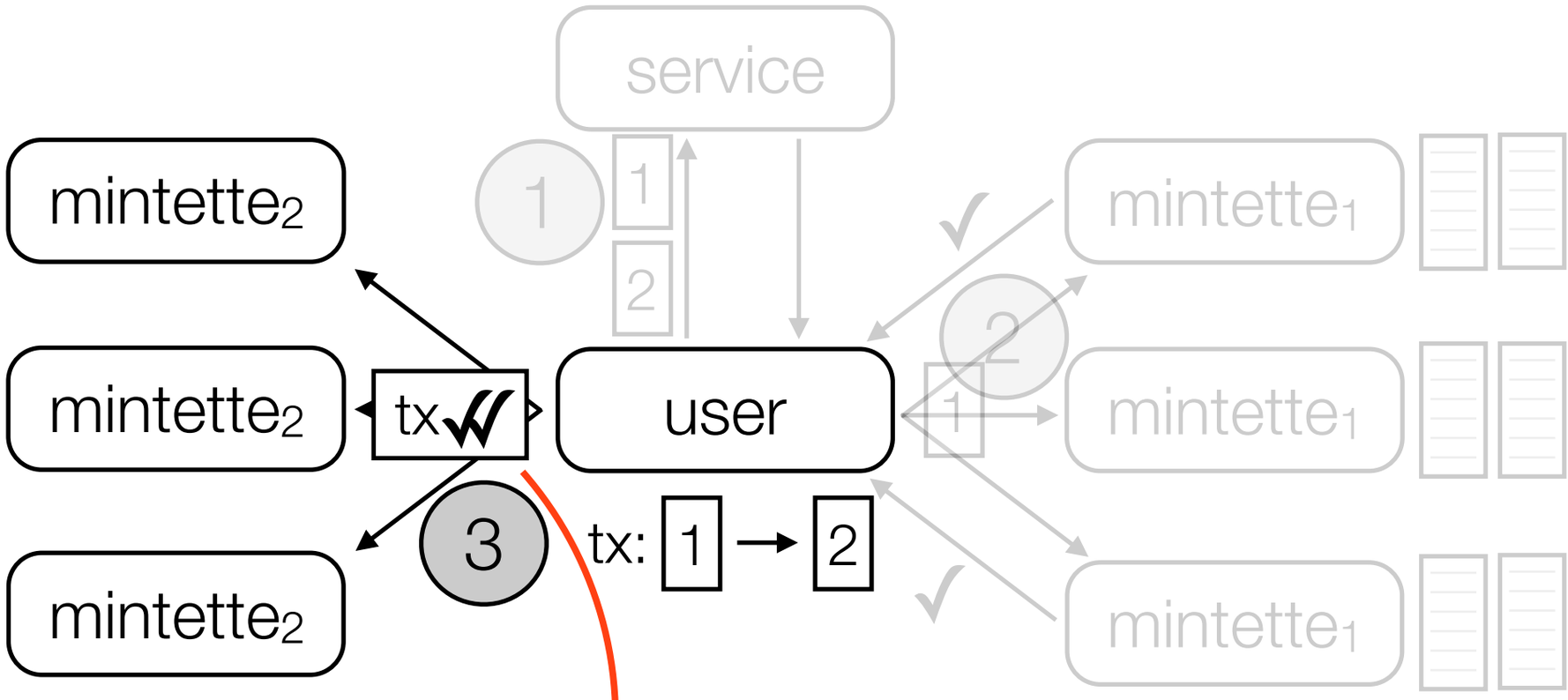
…using lists of **unspent transaction outputs** (utxo)


signed ‘yes’ vote
service
1
1
2
user
tx:
1
2
2
1
mintette1
mintette1
mintette1



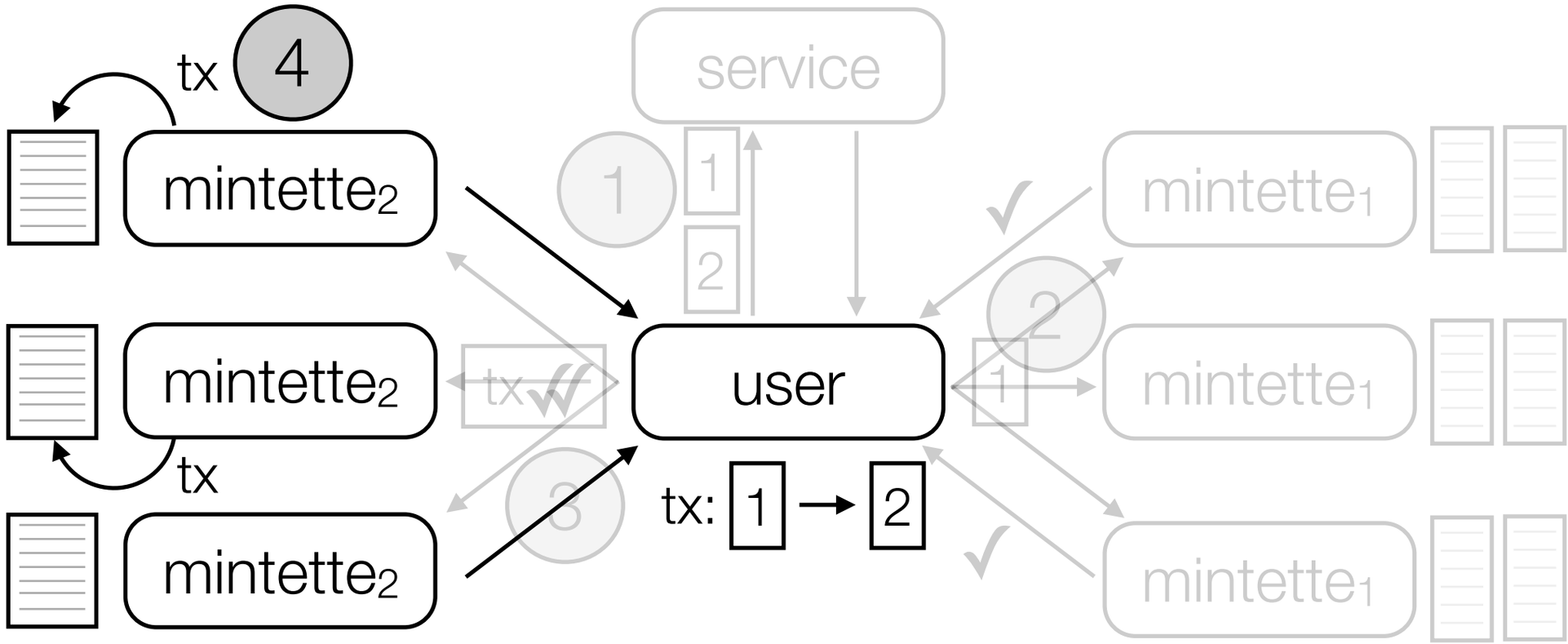
“bundle of evidence” contains ‘yes’ votes
from **majority** of mintettes in shard

mintettes check validity of bundle by checking for signatures from authorized mintettes…



“bundle of evidence” contains ‘yes’ votes from **majority** of mintettes in shard

…and if satisfied they add transaction
to be **committed** and send back **receipt**



consensus features

conceptually simple

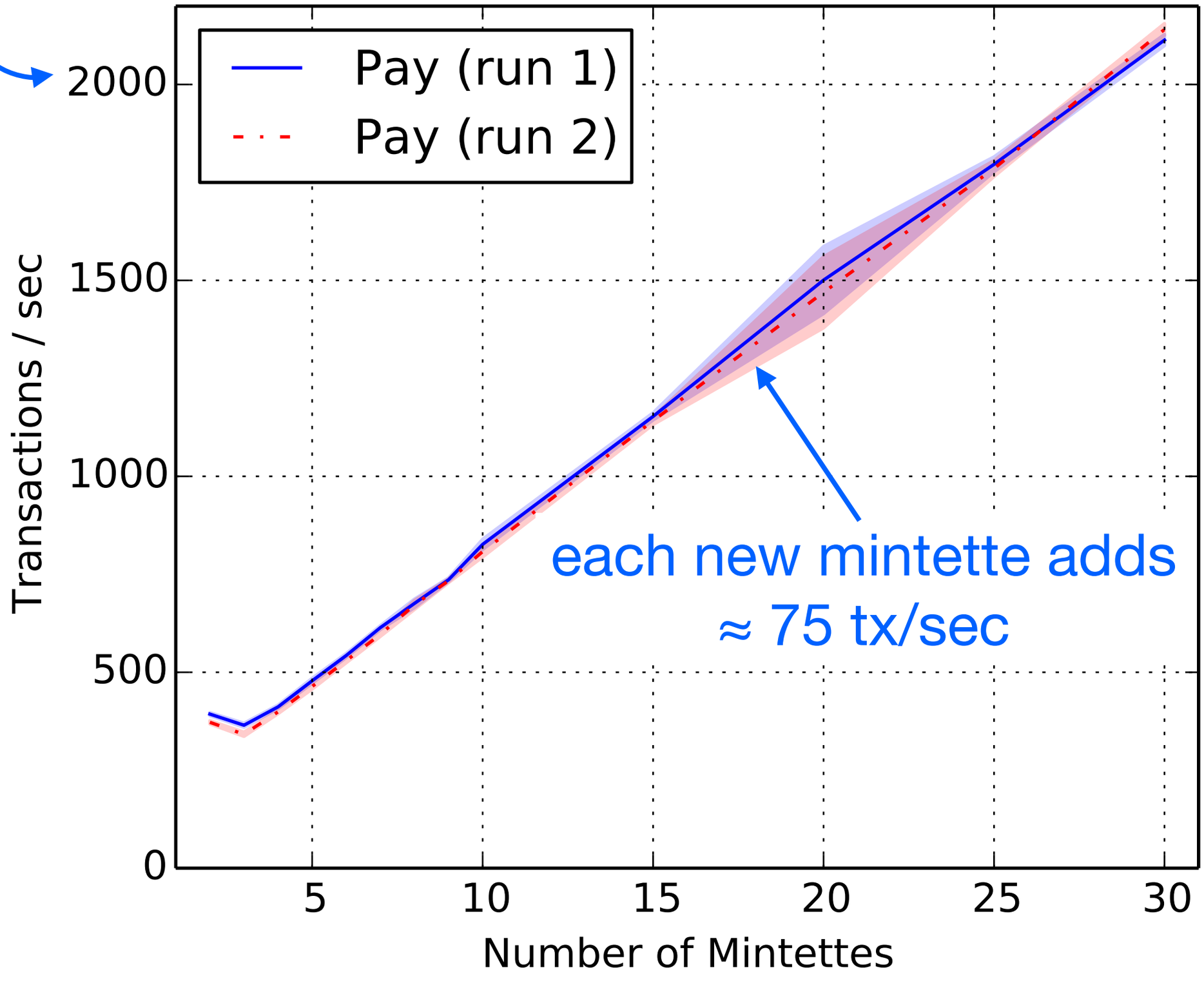
scalable!

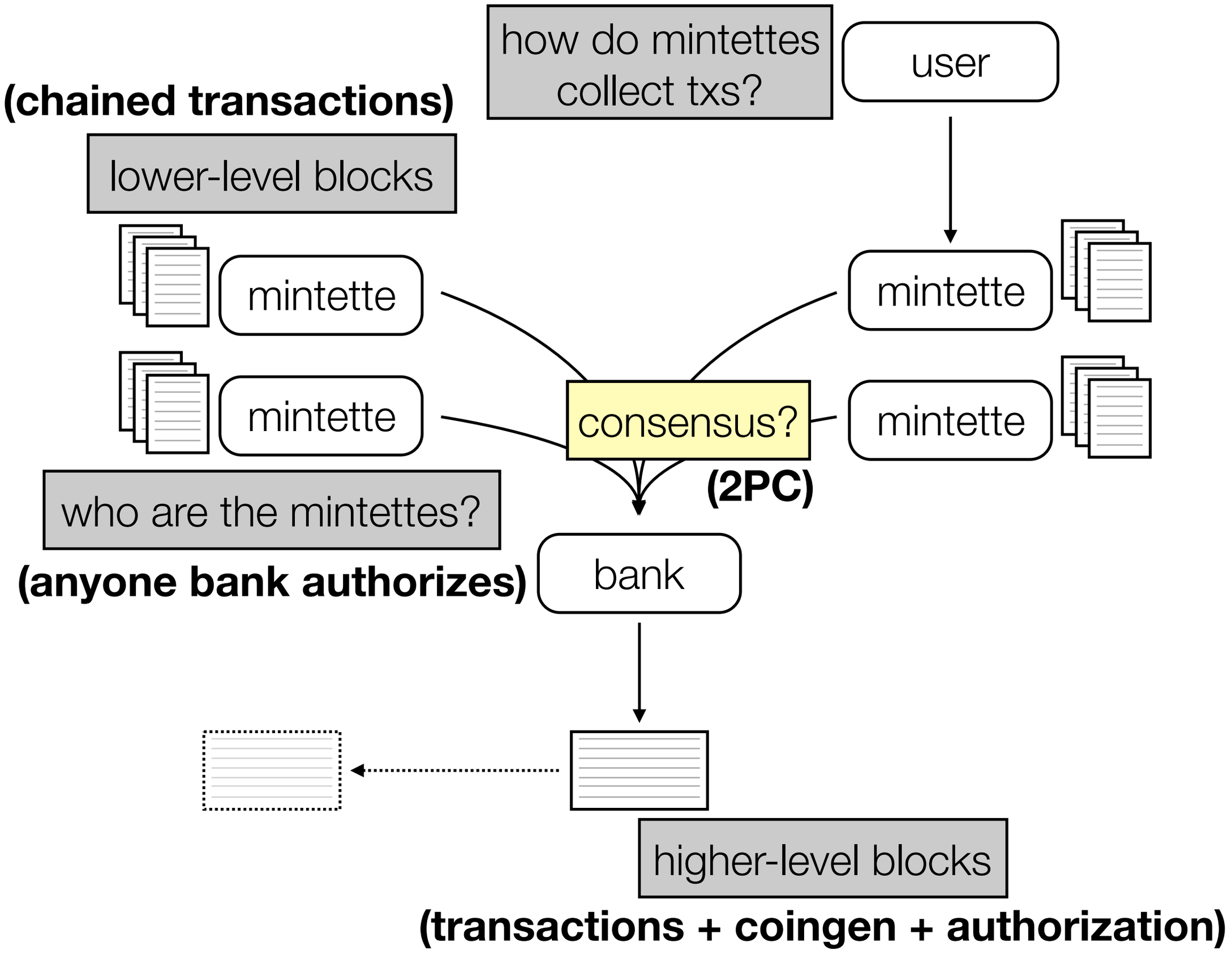
T = set of txs generated per second
Q = # mintettes per shard
M = # mintettes

$$\text{comm. per mintette per sec} = \frac{\sum_{tx \in T} 2(m_{tx}+1)Q}{M}$$

scales infinitely as more mintettes are added!

compared to Bitcoin's 7




how do mintettes collect txs?
user
(chained transactions)
lower-level blocks
mintette
mintette
mintette
mintette
consensus?
(2PC)
who are the mintettes?
(anyone bank authorizes)
bank
higher-level blocks
(transactions + coingen + authorization)

(contacted based on shard)

how do mintettes collect txs?

user

(chained transactions)

lower-level blocks

mintette

mintette

consensus?

mintette

mintette

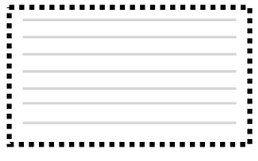
mintette

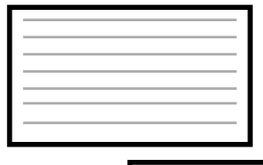
(2PC)

who are the mintettes?

(anyone bank authorizes)

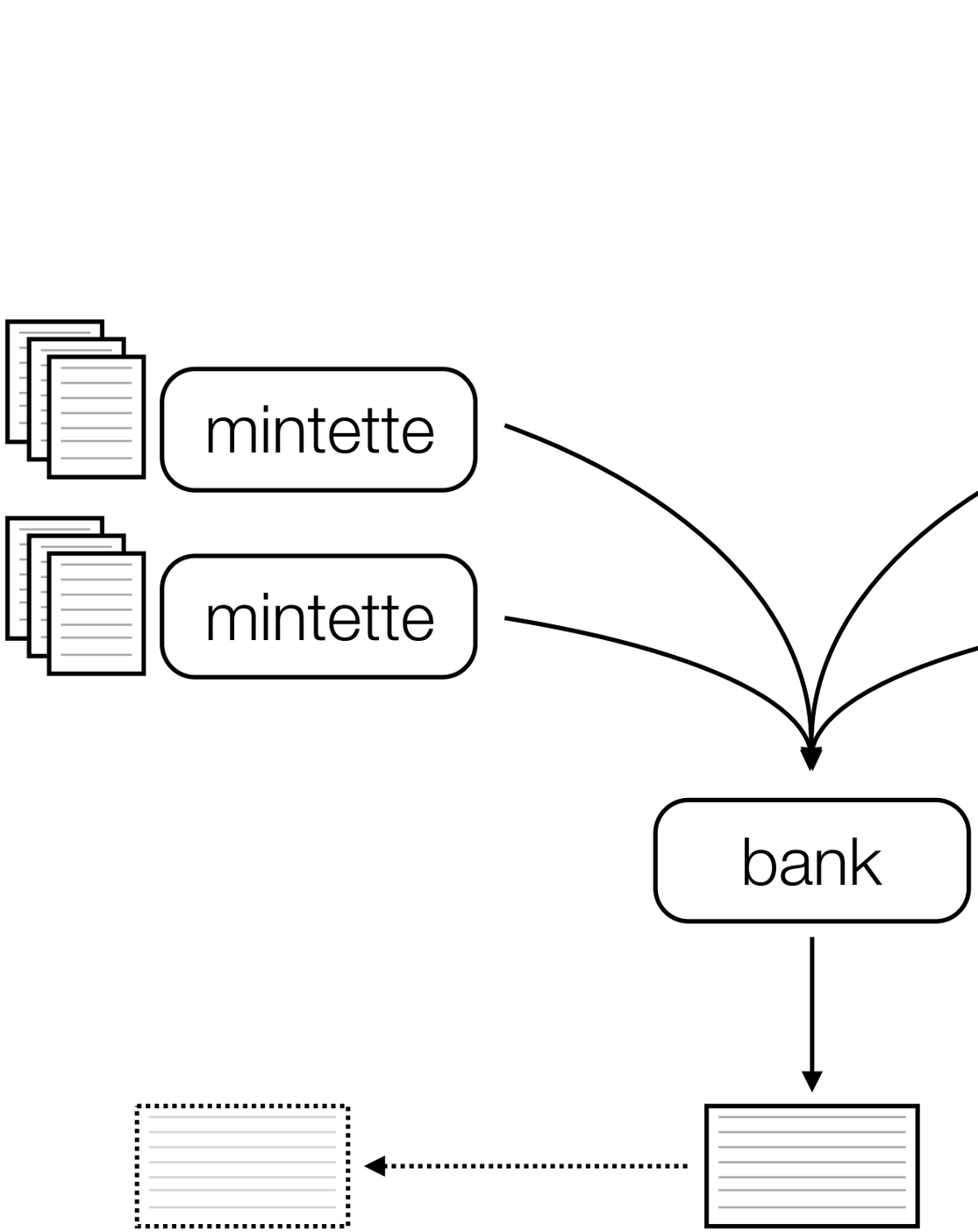
bank





higher-level blocks

(transactions + coingen + authorization)



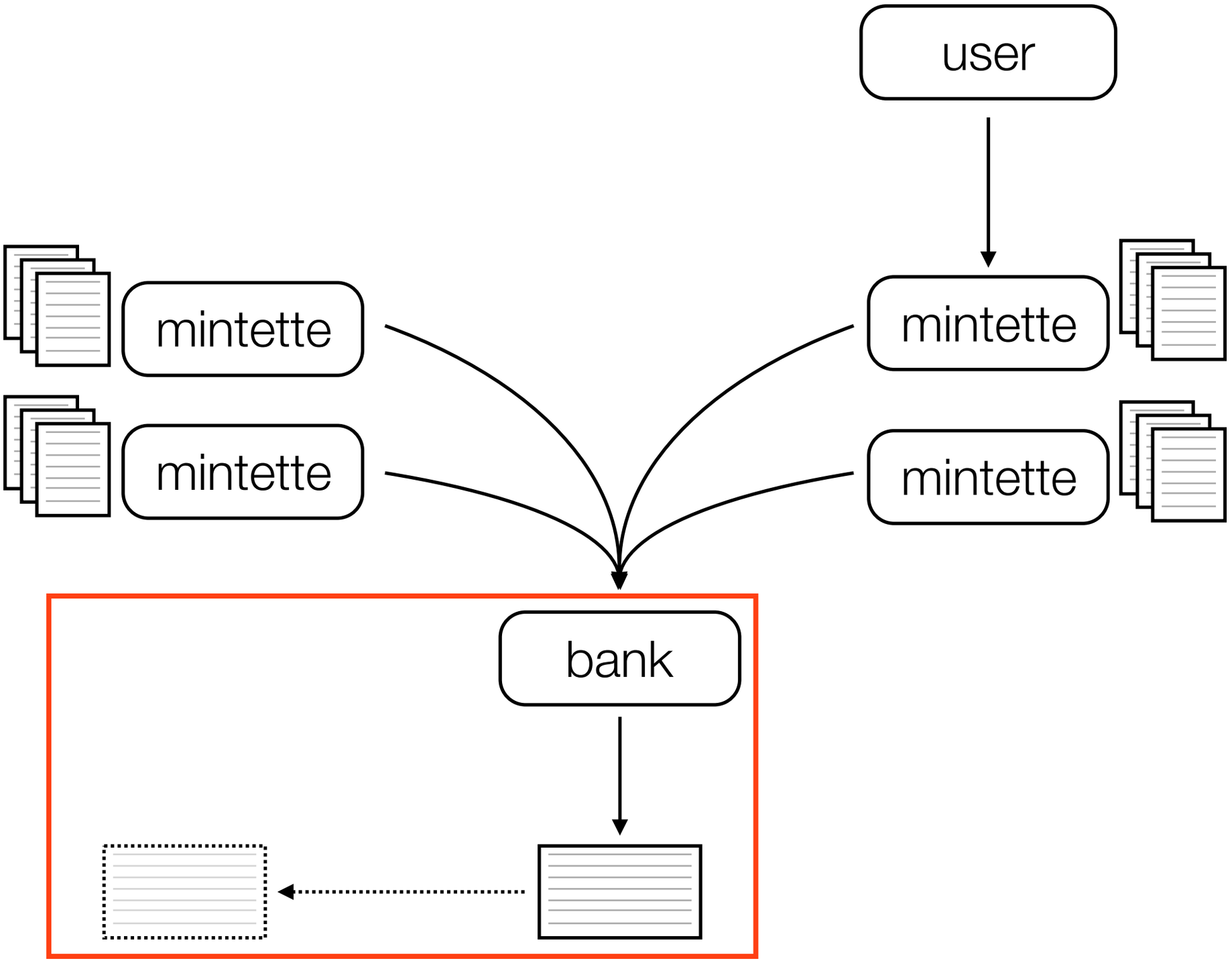
-collate transactions

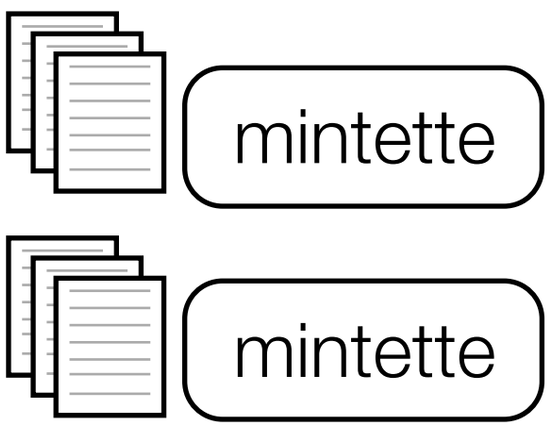
-allocate fees

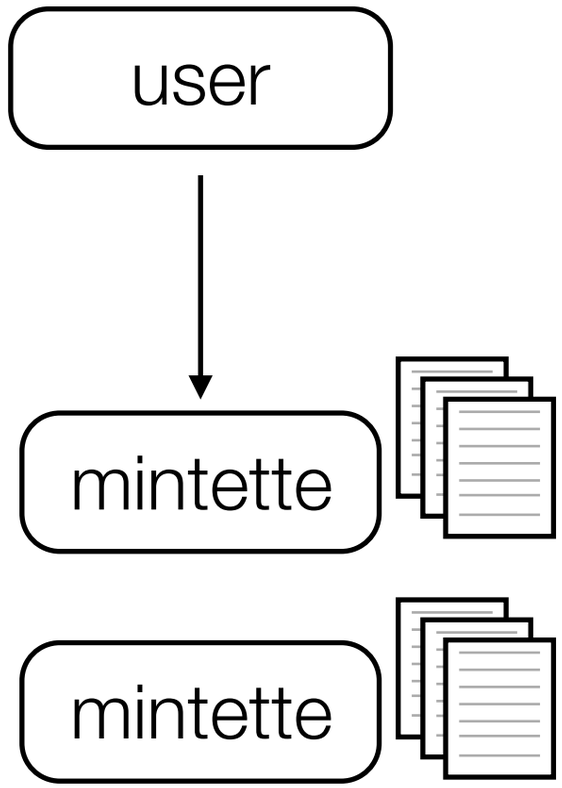
(-add coin generation)

-audit mintettes

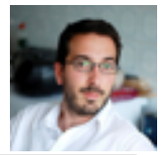
-authorize mintettes

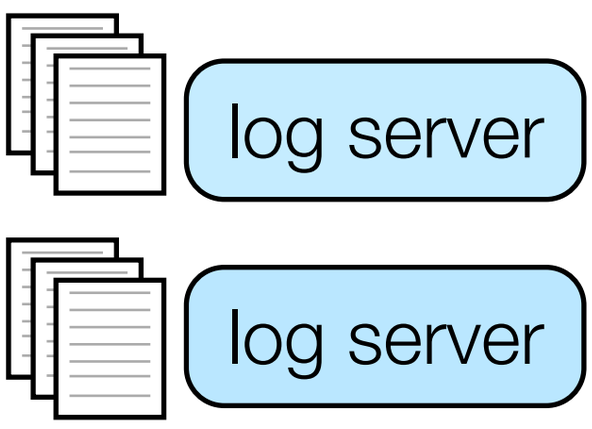

user
mintette
mintette
mintette
mintette
bank

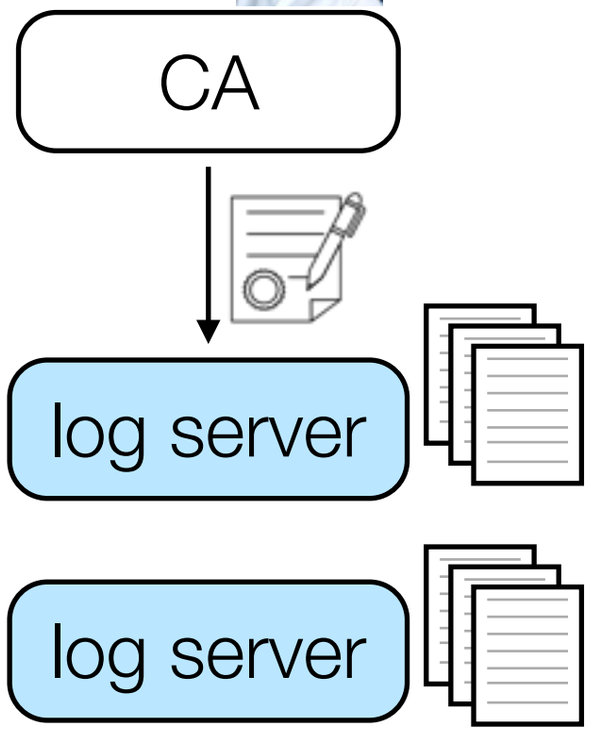

mintette
mintette


user
mintette
mintette

Certificate Transparency

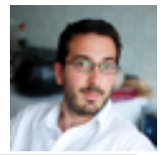






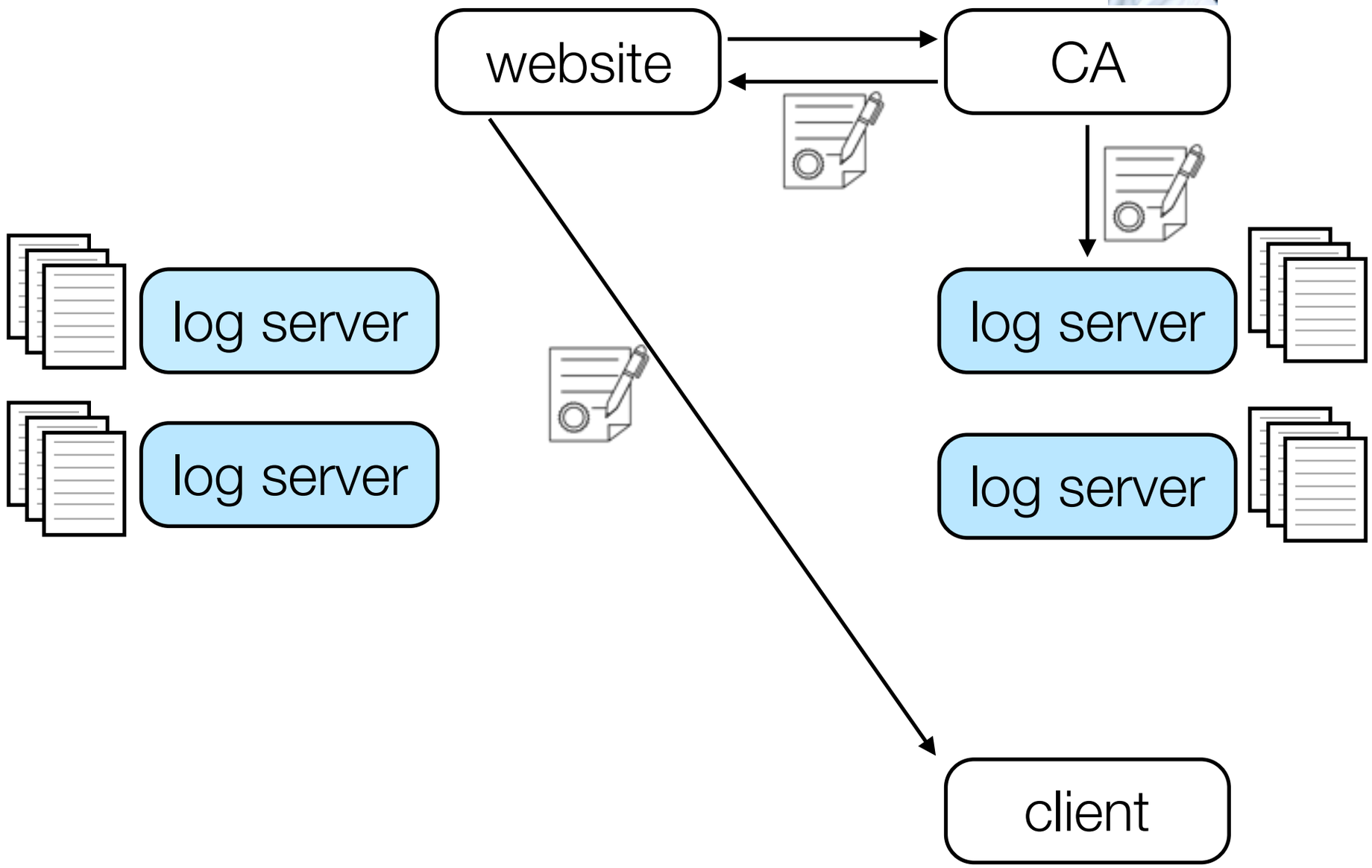
(icon by parkjisun for the noun project)

Certificate Transparency



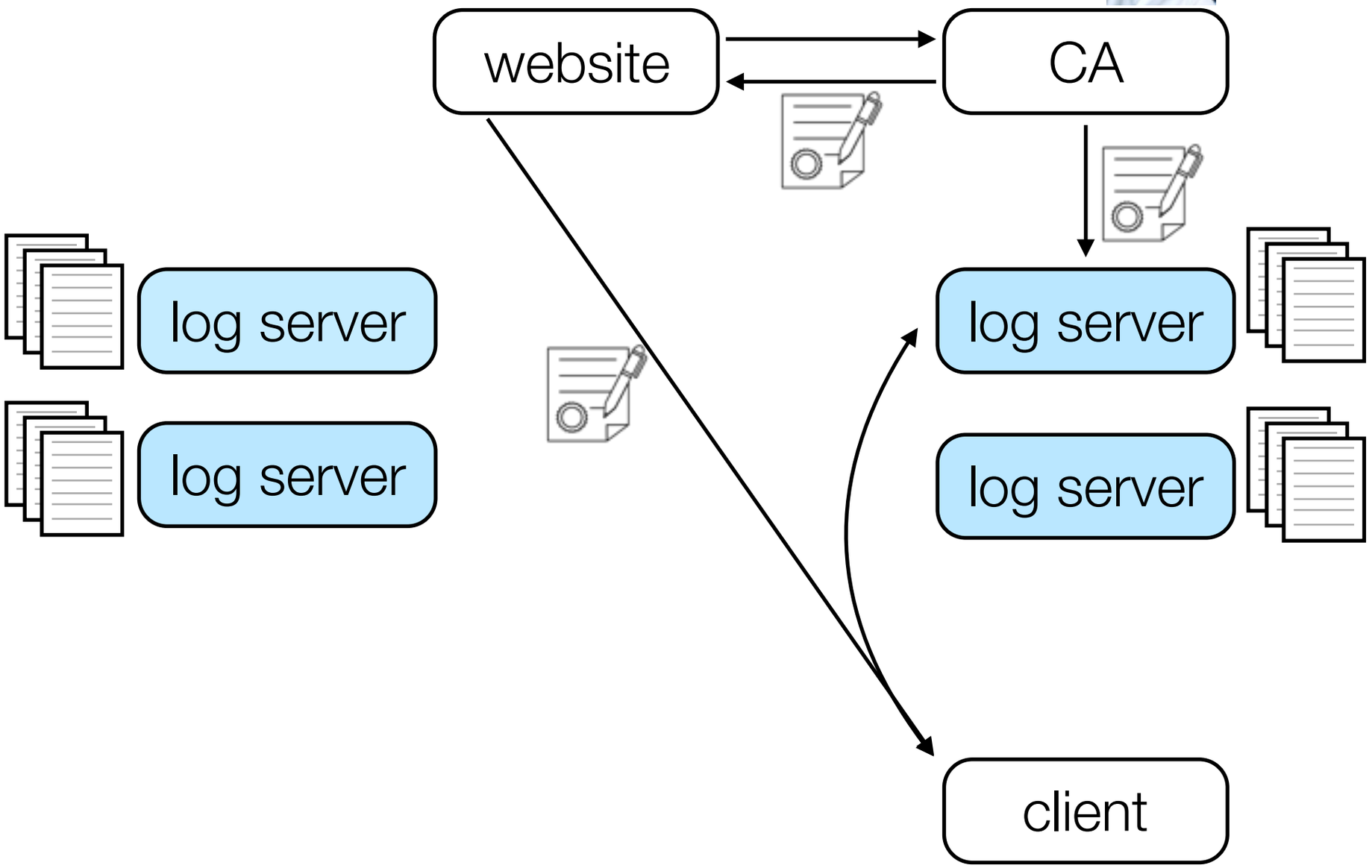
(icon by parkjisun for the noun project)

Certificate Transparency



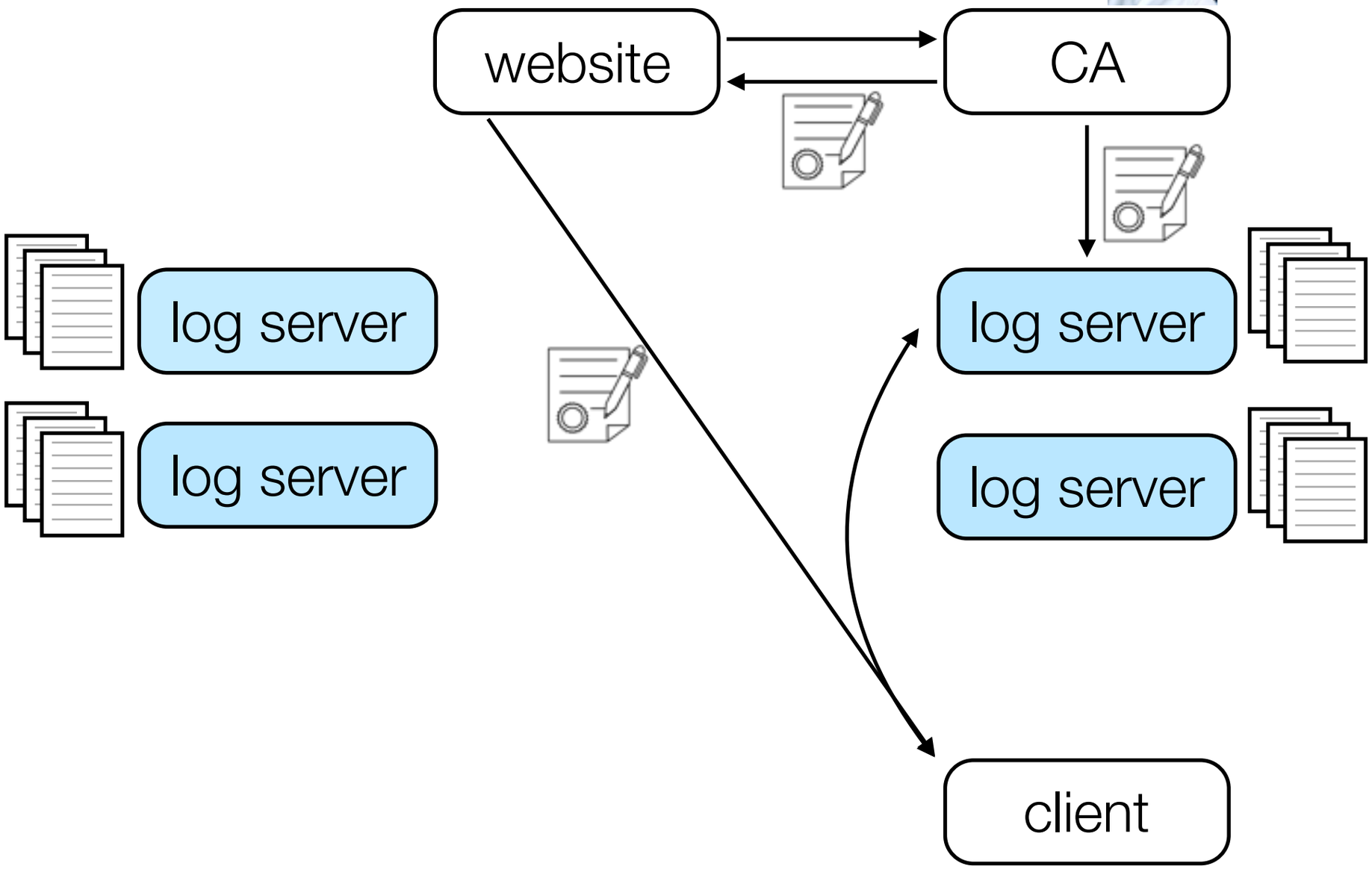
(icon by parkjisun for the noun project)

Certificate Transparency



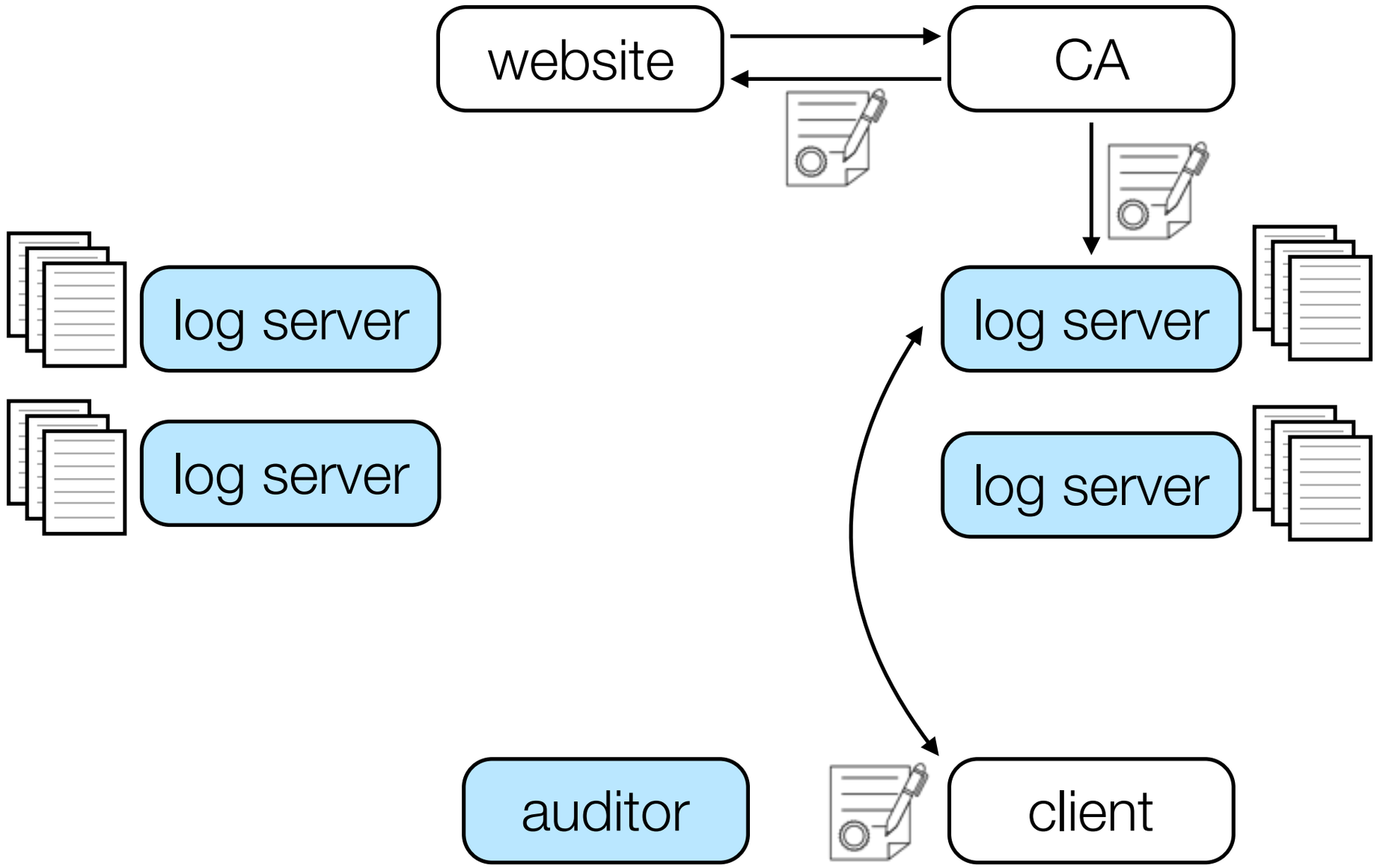
(icon by parkjisun for the noun project)

Certificate Transparency

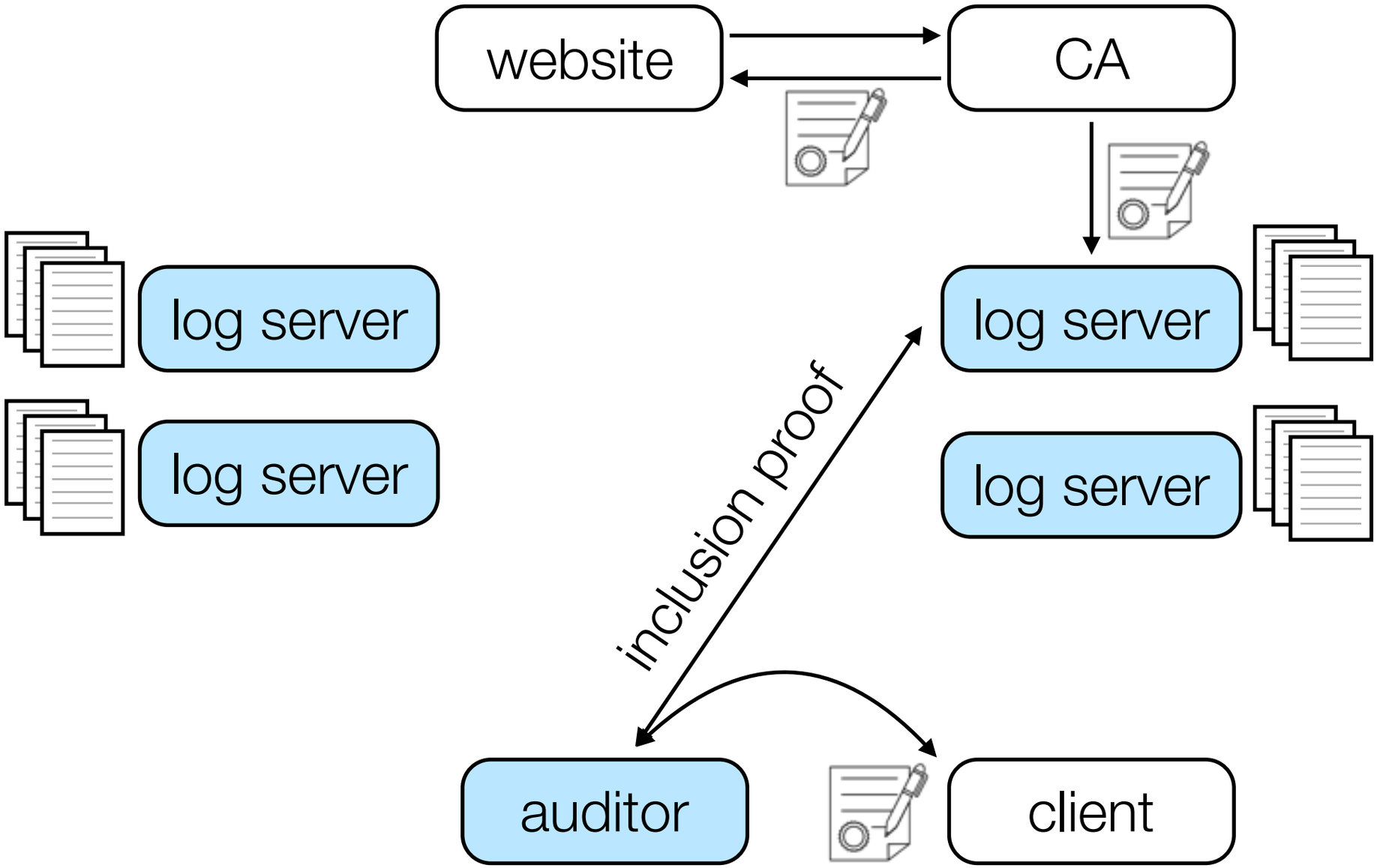


goal: don’t let clients accept bad certificates

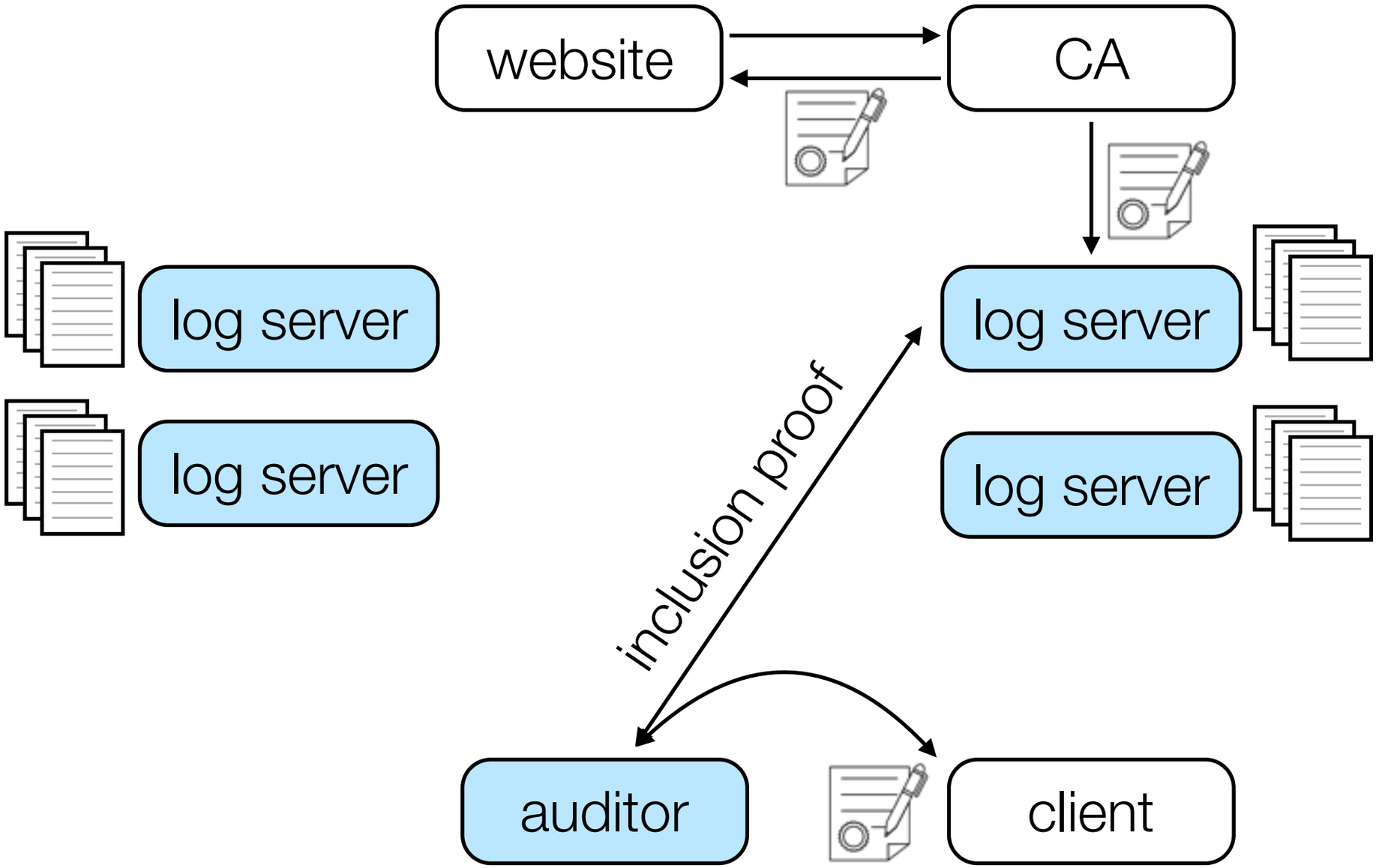
(icon by parkjisun for the noun project)



(icon by parkjisun for the noun project)

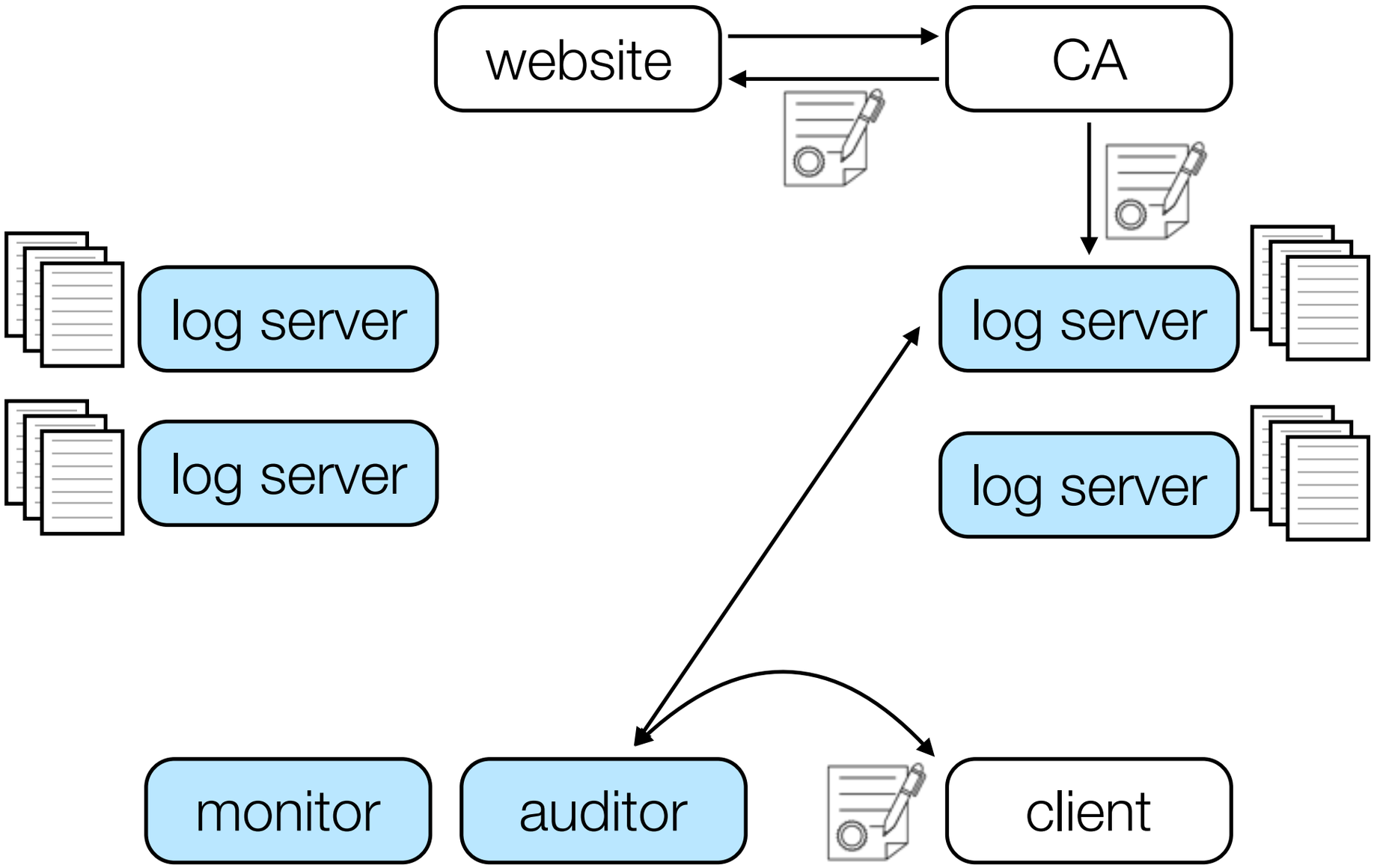


(icon by parkjisun for the noun project)

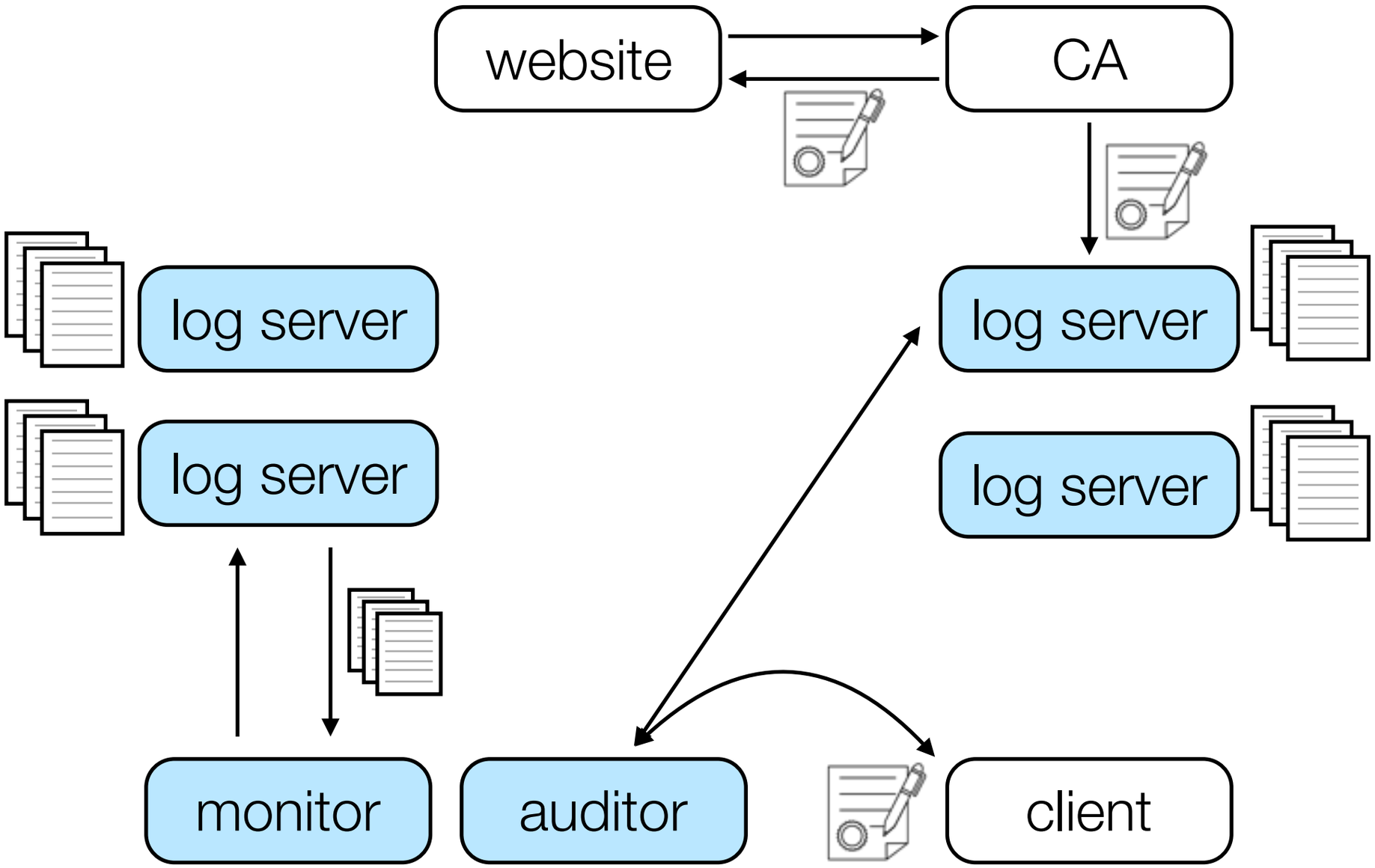


auditors (efficiently) determine if certificates are in the log

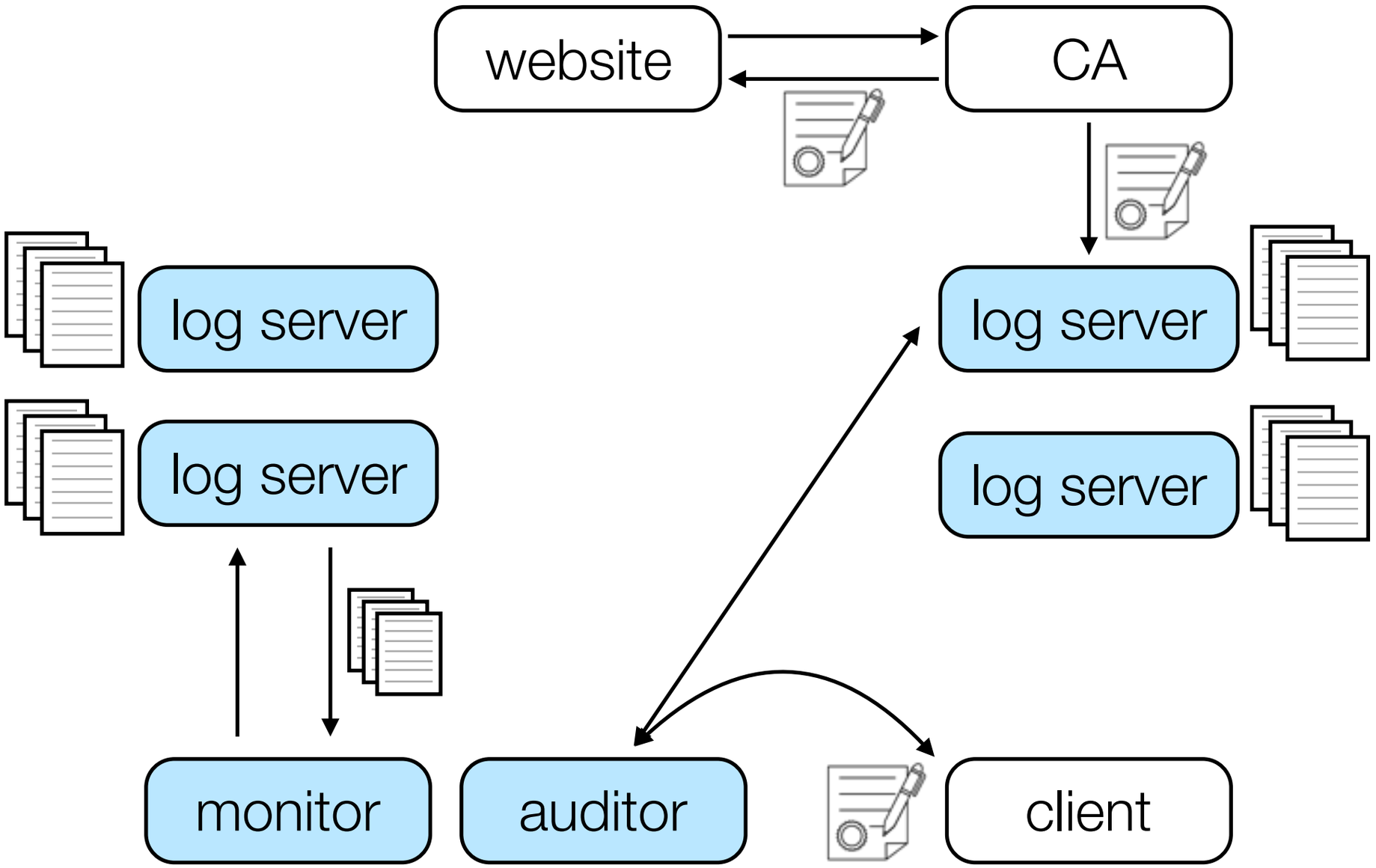
(icon by parkjisun for the noun project)



(icon by parkjisun for the noun project)

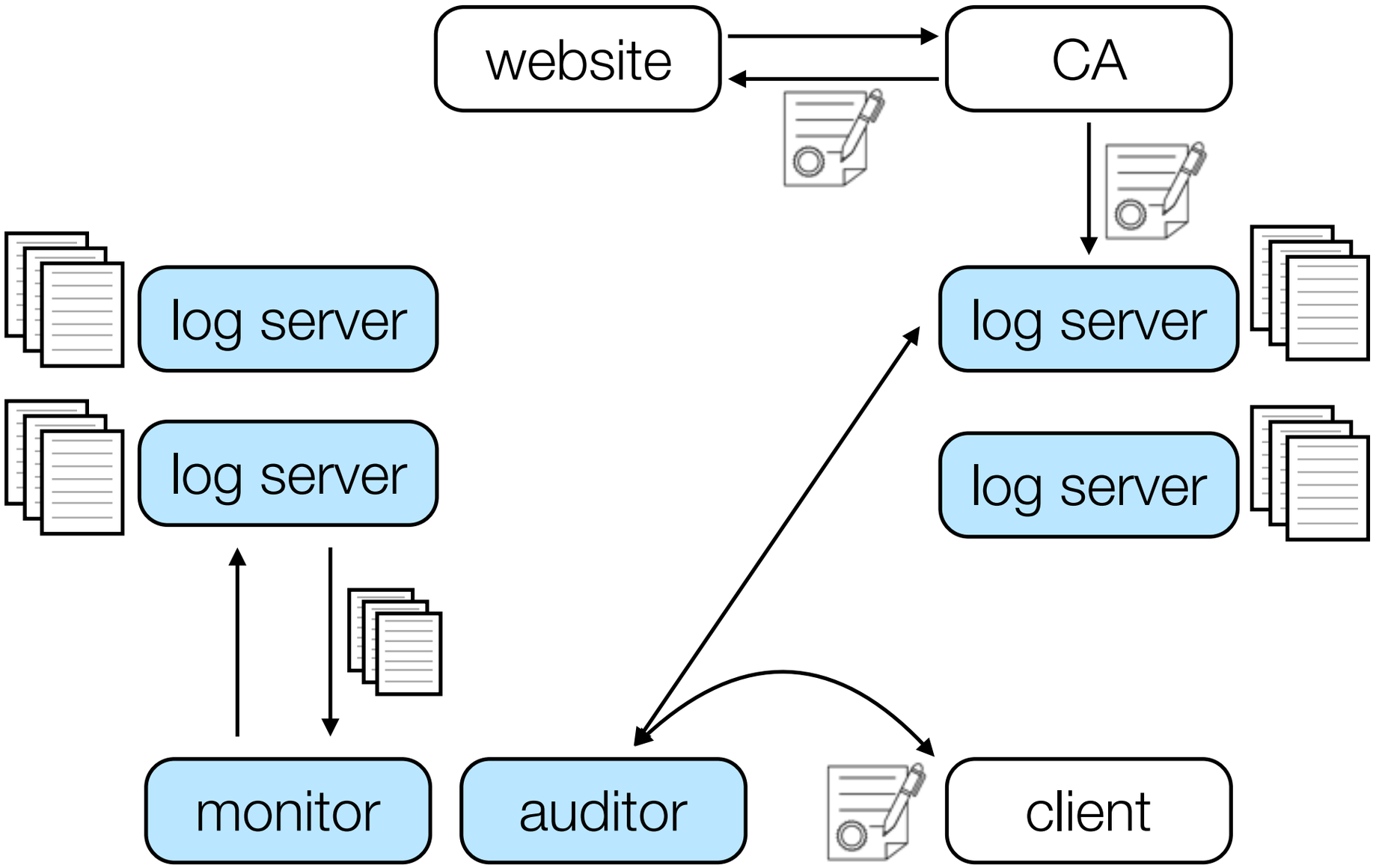


(icon by parkjisun for the noun project)

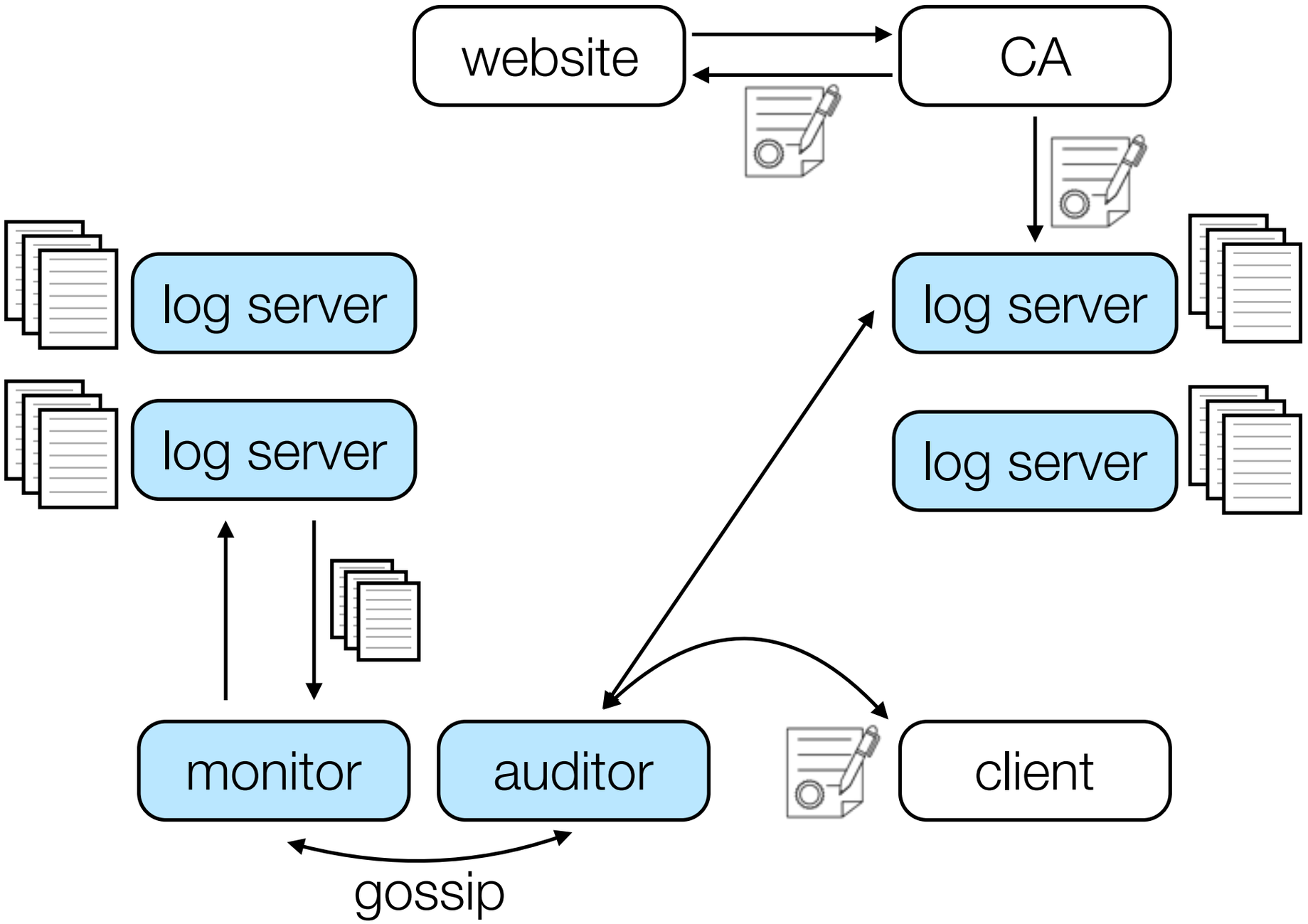


monitors (inefficiently) detect bad certificates in the log

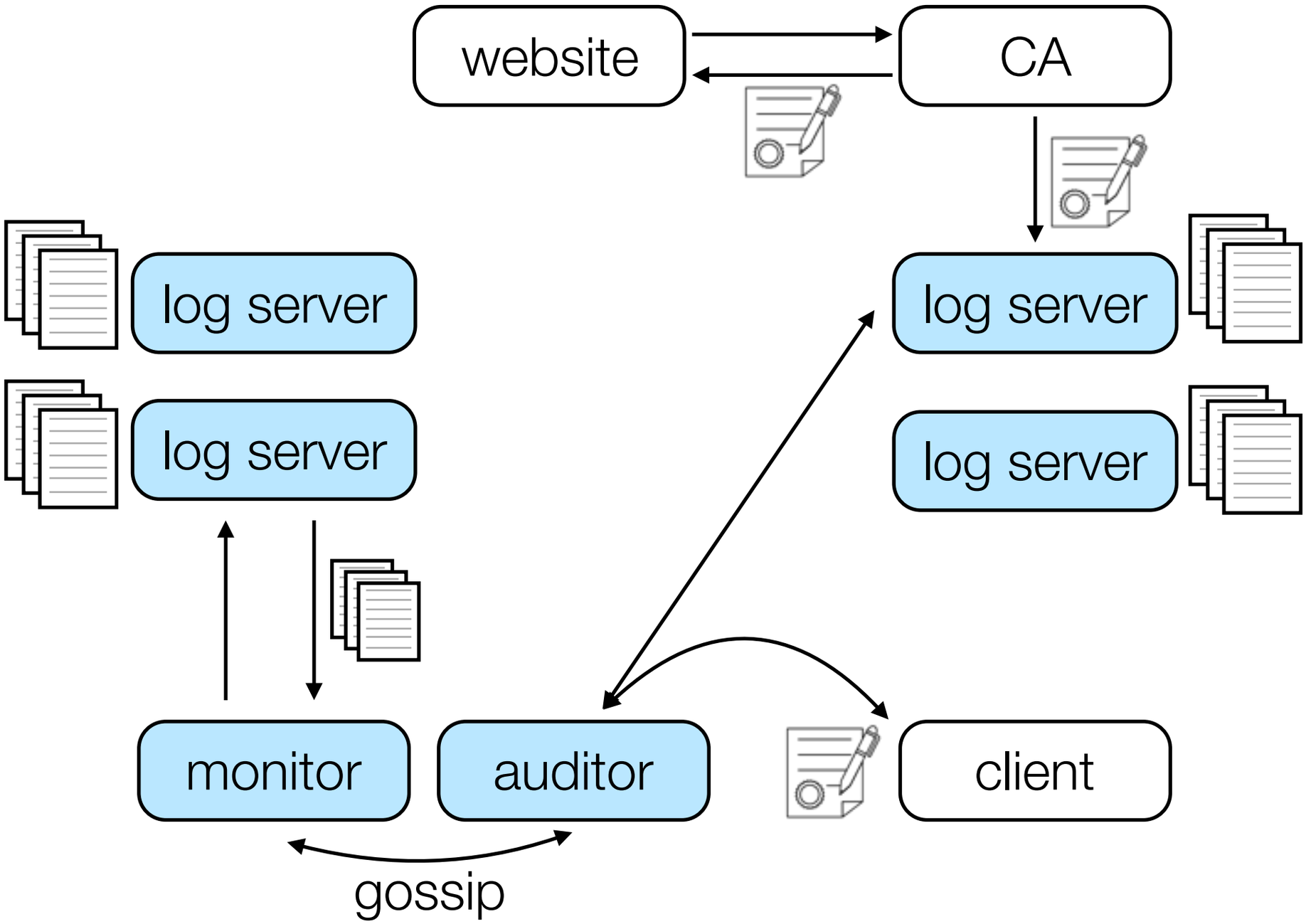
(icon by parkjisun for the noun project)



(icon by parkjisun for the noun project)



(icon by parkjisun for the noun project)

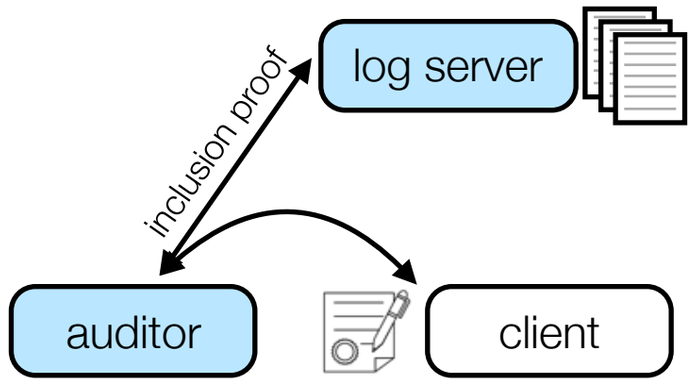


auditors and monitors ensure consistent view of log

(icon by parkjisun for the noun project)

goal: don’t let clients accept bad certificates

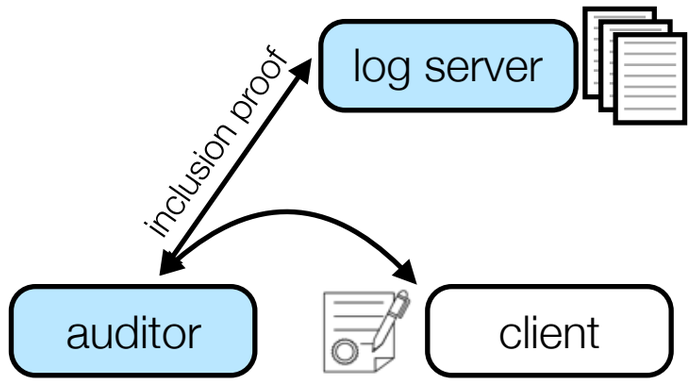
(icon by parkjisun for the noun project)



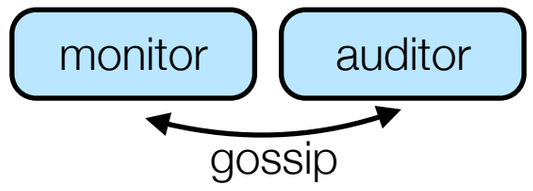
auditors (efficiently) determine if certificates are in the log

goal: don’t let clients accept bad certificates

(icon by parkjisun for the noun project)



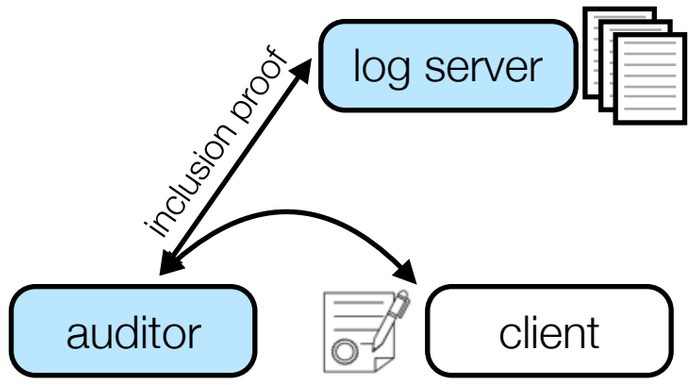
auditors (efficiently) determine if certificates are in the log



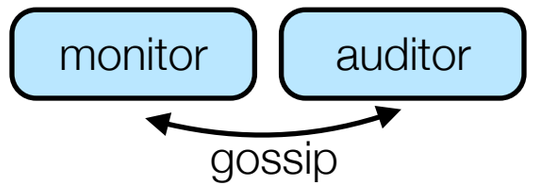
+ auditors and monitors ensure consistent view of log

goal: don’t let clients accept bad certificates

(icon by parkjisun for the noun project)



auditors (efficiently) determine if certificates are in the log

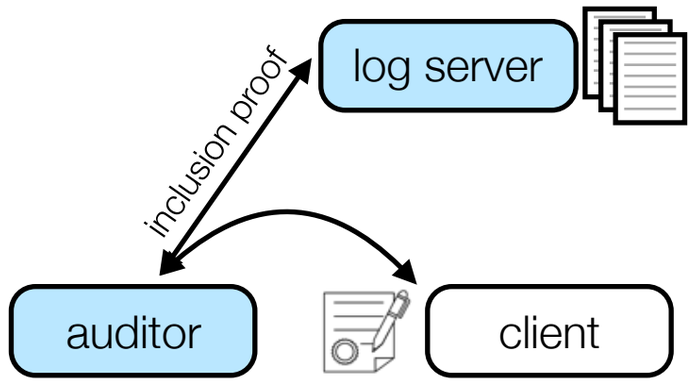


+ auditors and monitors ensure consistent view of log

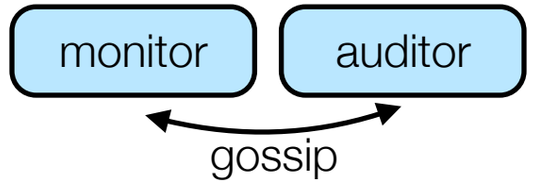
⇒ certificate is in monitor's view of the log

goal: don't let clients accept bad certificates

(icon by parkjisun for the noun project)

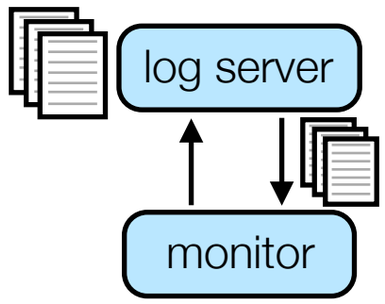


auditors (efficiently) determine if certificates are in the log



+ auditors and monitors ensure consistent view of log

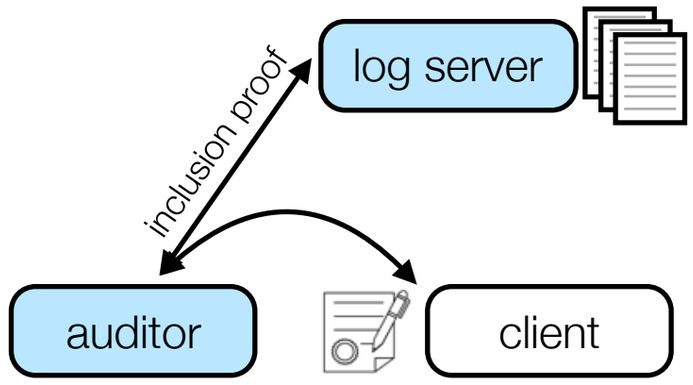
⇒ certificate is in monitor's view of the log



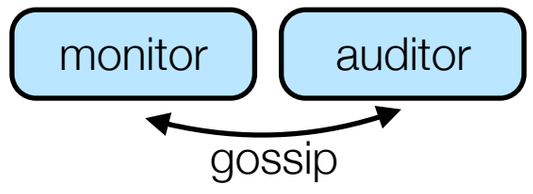
+ monitors (inefficiently) detect bad certificates in the log

goal: don't let clients accept bad certificates

(icon by parkjisun for the noun project)

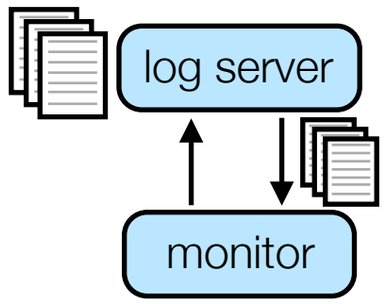


auditors (efficiently) determine if certificates are in the log



+ auditors and monitors ensure consistent view of log

⇒ certificate is in monitor's view of the log

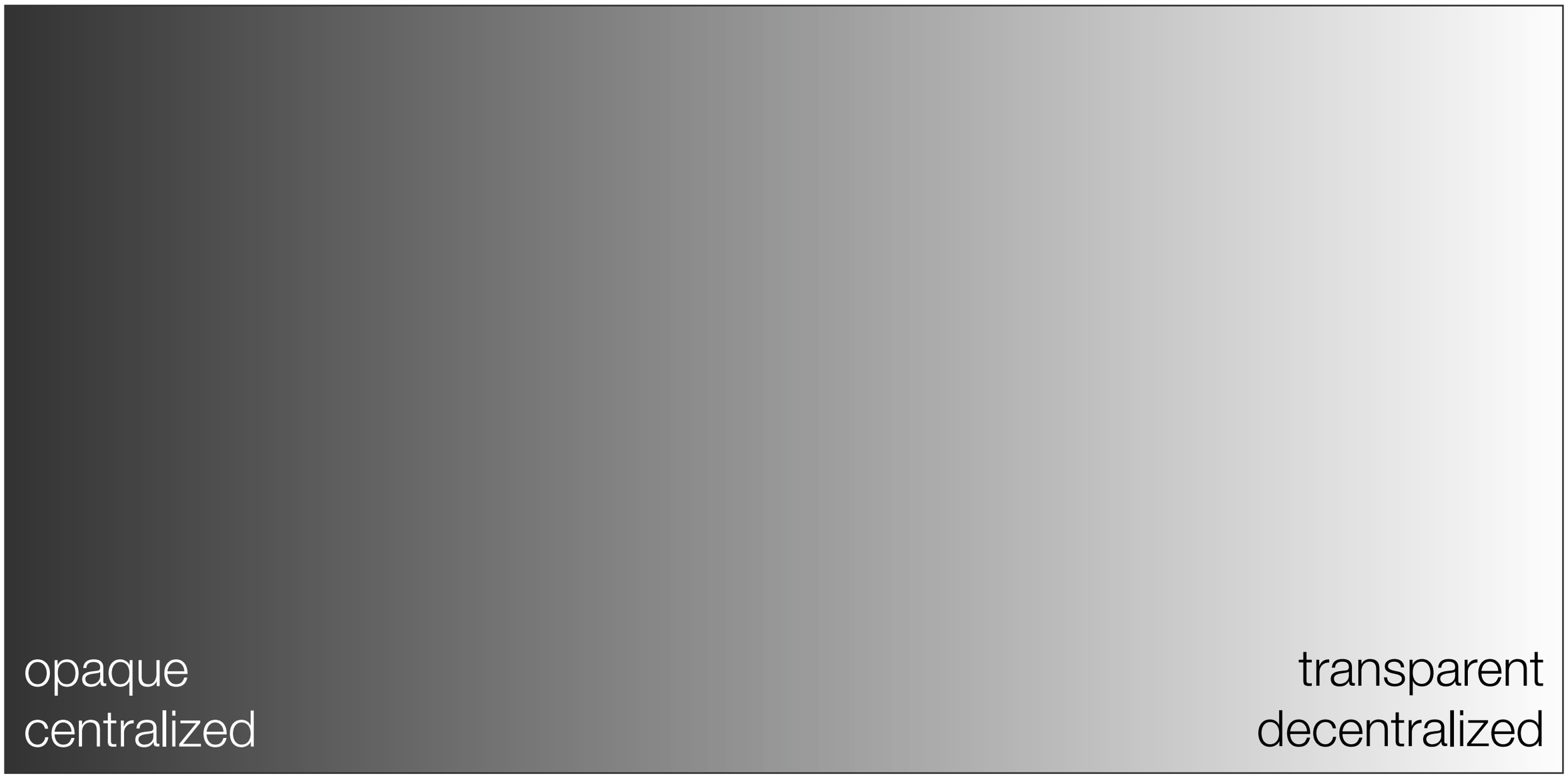


+ monitors (inefficiently) detect bad certificates in the log

⇒

goal: don't let clients accept bad certificates

(icon by parkjisun for the noun project)


opaque
centralized
transparent
decentralized

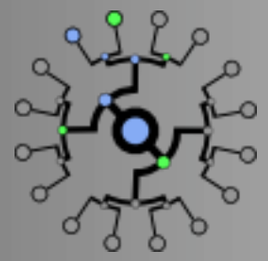
opaque
centralized

transparent
decentralized

opaque
centralized

transparent
decentralized

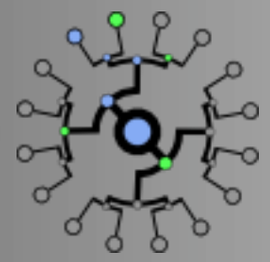
RSCoin



opaque
centralized

transparent
decentralized

RSCoin

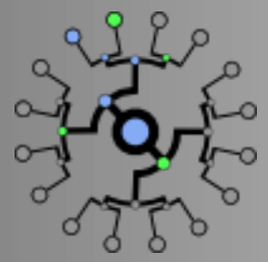


opaque
centralized

transparent
decentralized

?

RSCoin

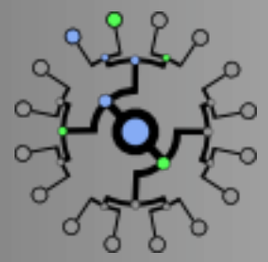


opaque
centralized

?

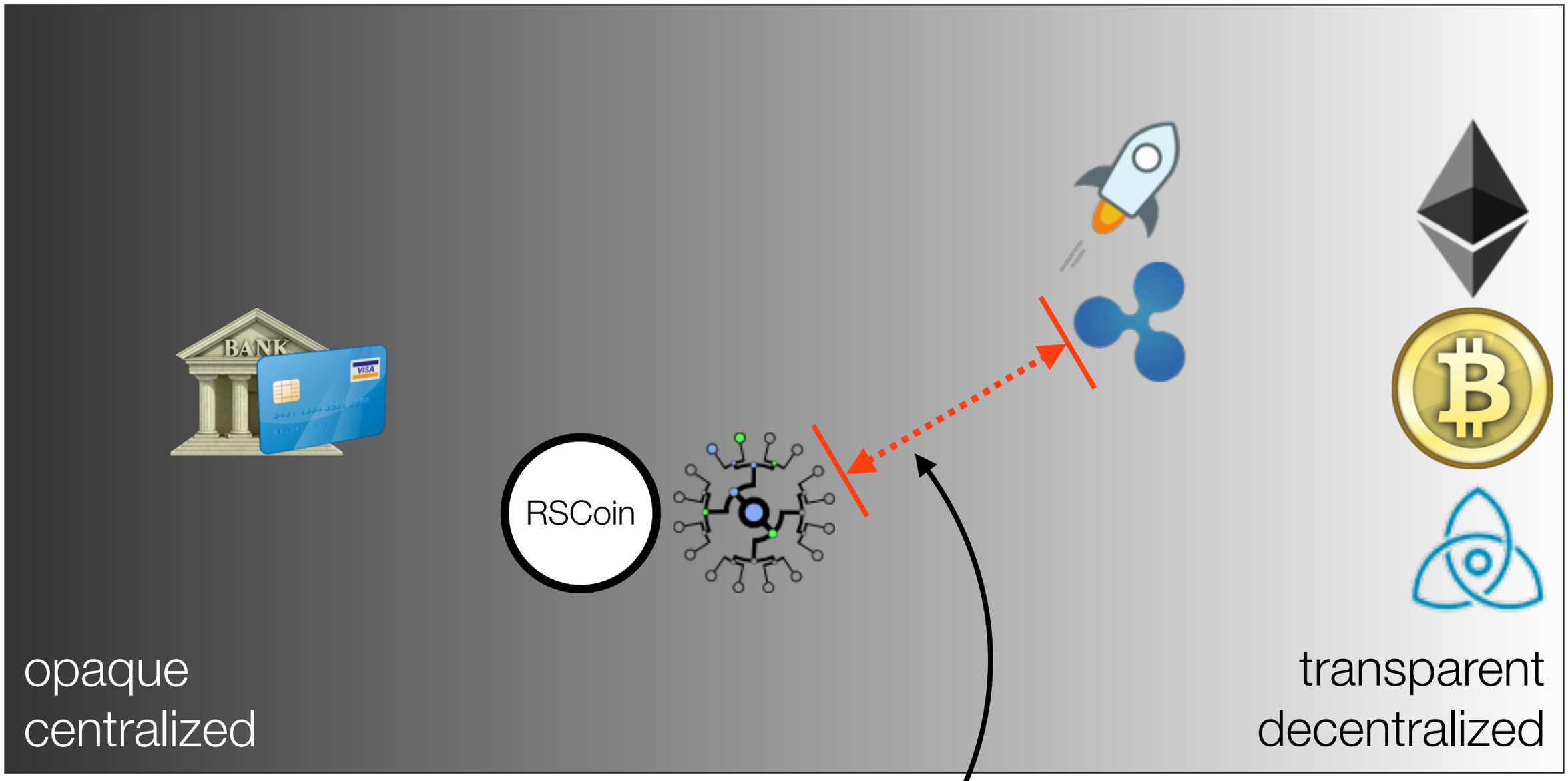
transparent
decentralized

RSCoin



opaque
centralized

transparent
decentralized

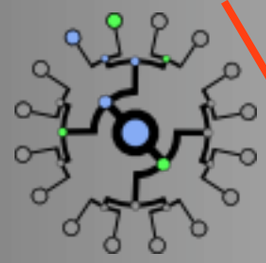


what is this distance? can we quantify it?

better consumer protection? better fairness?

absolute accountable non-existent

RSCoin



opaque
centralized

transparent
decentralized

what is this distance? can we quantify it?

what security properties are even provided?

better consumer protection? better fairness?

absolute accountable non-existent

Thanks for listening!

opaque
centralized

transparent
decentralized

what is this distance? can we quantify it?

what security properties are even provided?