MicroCash: Practical Concurrent Processing of Micropayments

Ghada Almashaqbeh[1], Allison Bishop[2], **Justin Cappos**[3]

[1]CacheCash, [2]Columbia and Proof Trading, [3]NYU

Financial Crypto, Malaysia, 2020

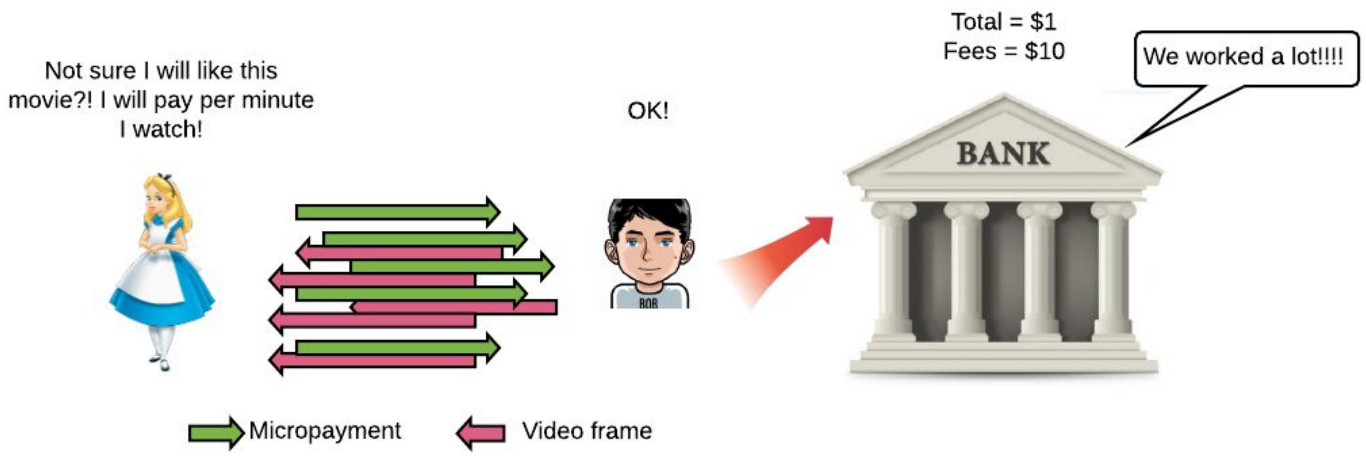
Outline

- Background.
- Motivation.
- MicroCash design.
    - Escrow setup.
    - Paying with lottery tickets.
    - Lottery protocol.
    - Claiming payments.
    - Proof of cheating and the penalty deposit.
- Security analysis.
- Performance evaluation.
- Conclusion.

Micropayments

"MICROPAYMENTS ARE BACK, AT LEAST IN THEORY, THANKS TO P2P" *

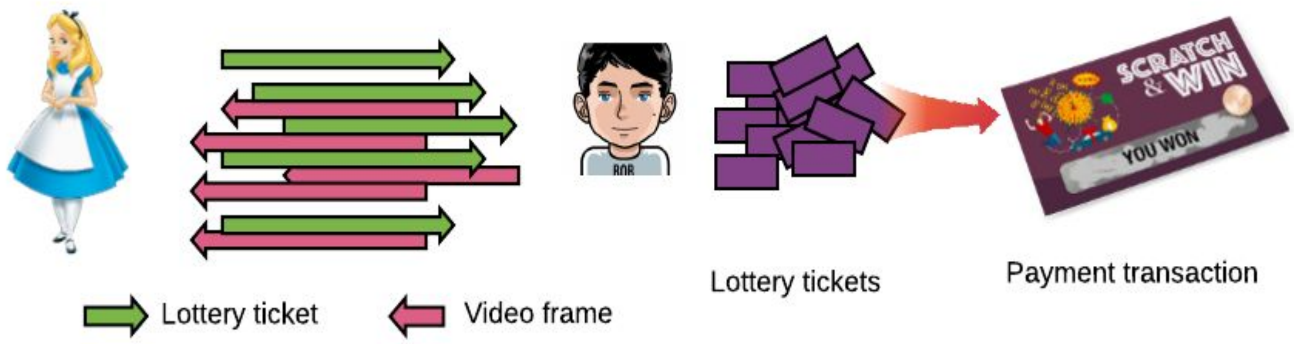
- Payments of micro values (pennies or fractions of pennies).
- Several potential applications.
  - Ad-free web surfing, online gaming, and rewarding peers in peer-assisted services.
- ***Drawbacks***; high transaction fees and large log size.



[*] Clay Shirky, The Case Against Micropayments, http://www.openp2p.com/pub/a/p2p/2000/12/19/micropayments.html

Probabilistic Micropayments

- A solution to aggregate tiny payments.
- Dated back to Rivest [Rivest, 1997] and Wheeler [Wheeler, 1996].



- Early implementations were centralized.
- Cryptocurrencies are utilized to achieve decentralization.

Decentralized Probabilistic Micropayments

- Ingredients:
  - Trusted bank ⇒ Miners.
  - Bank accounts to hold payments ⇒ Escrows on the blockchain.
  - Distributed lottery protocol.
- Main challenges:
  - Ticket duplication (pay several parties the same lottery ticket).
  - Front running attacks.
- Prior work.
  - Only two schemes: **MICROPAY** [Pass et al., 2015] and **DAM** [Chiesa et al., 2017]

Prior Work Limitations

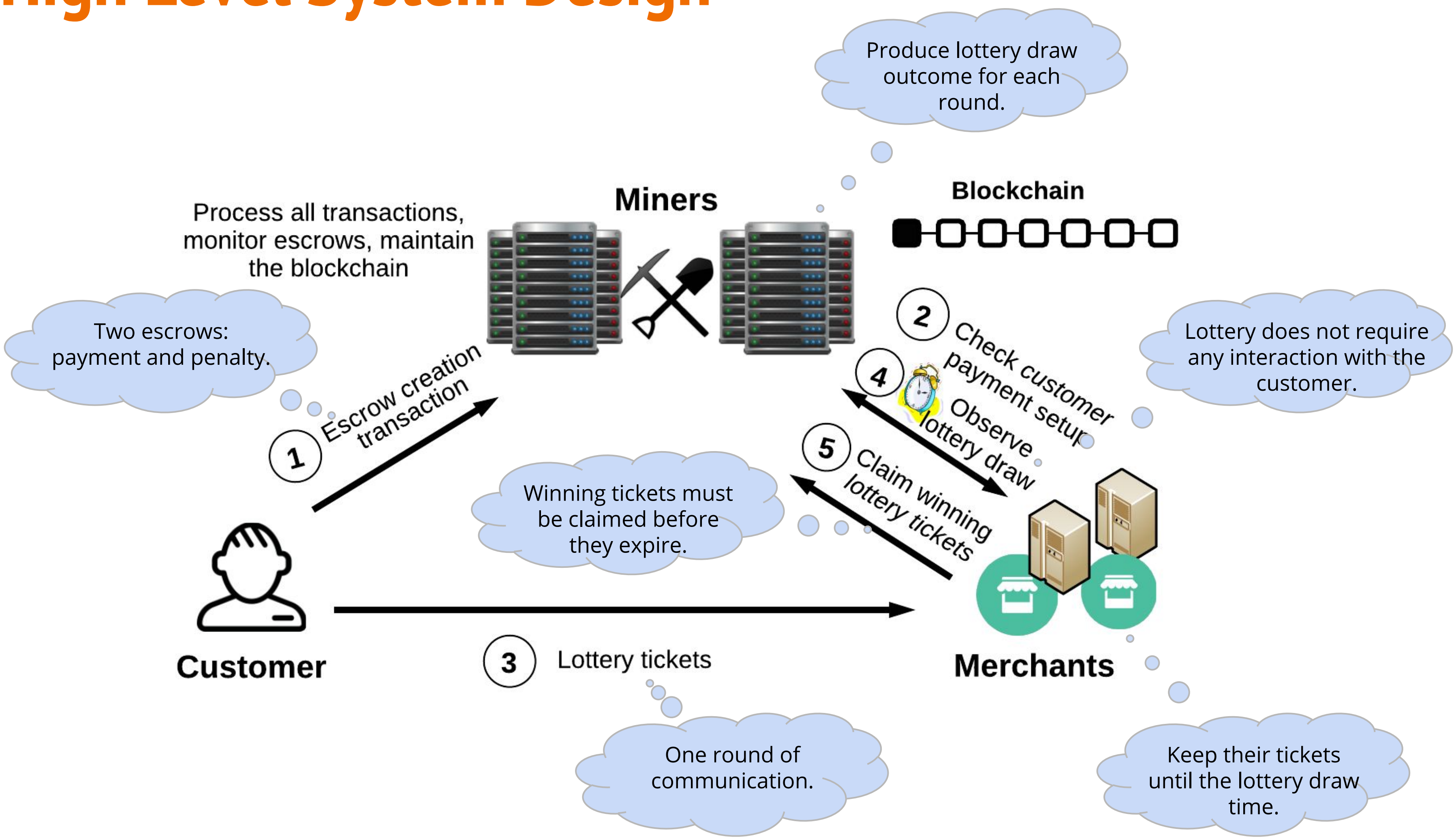
- Support only **sequential** micropayments.
    - High latency, large number of escrows (more fees and larger blockchain size).
- Interactive lottery protocol.
    - Require **several rounds** of communication to exchange a lottery ticket.
- Chances of having all, or no, tickets win.
    - Psychological obstacle as a customer may pay more than expected.
- Computationally-heavy.

MicroCash addresses these limitations!!

MicroCash Overview

- The ***first*** decentralized probabilistic micropayment scheme that supports **concurrent micropayments**.
- Requires ***one*** round of communication to exchange a ticket.
    - Introduces a non-interactive and lightweight lottery protocol based solely on secure hashing.
- The ***first*** to introduce a lottery protocol with ***exact win rate.***
- Reduces the amount of data to be logged on the blockchain by around **50%** (compared to sequential micropayment schemes).
- Increases ticket processing rate by **1.7 - 4.2x** (compared to MICROPAY).

High Level System Design



Escrows and Micropayment Concurrency

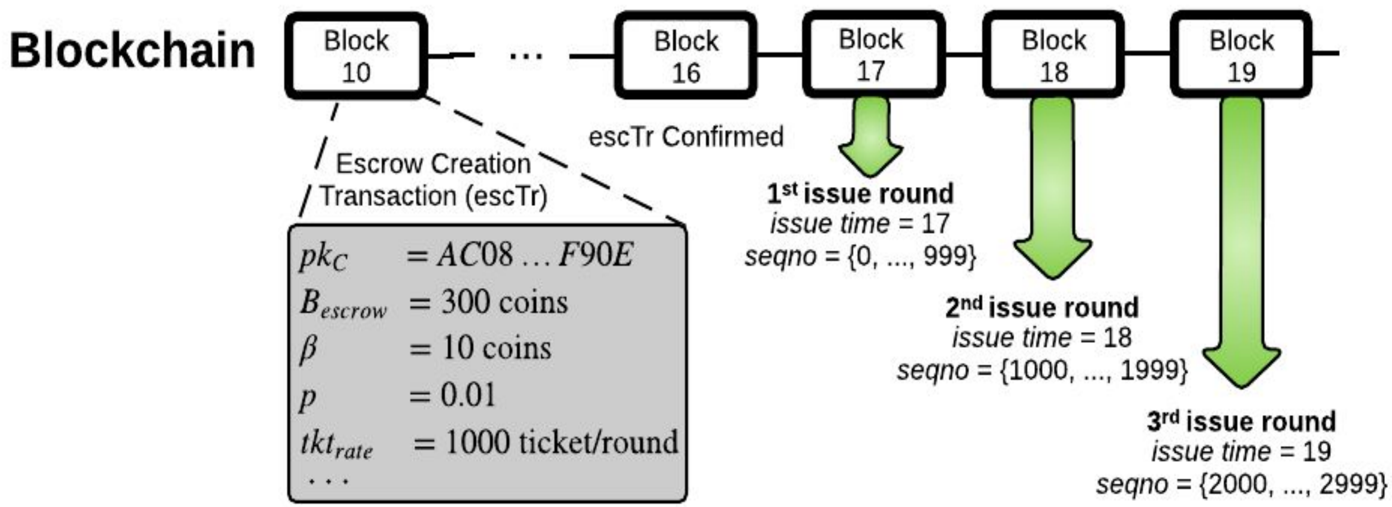
- The payment escrow balance covers all winning tickets.
  - A winning probability $\boldsymbol{p}$, ticket issue rate $\boldsymbol{tkt_{rate}}$, lottery round length $\boldsymbol{draw_{len}}$, and escrow lifetime $\boldsymbol{l_{esc}}$.
  - Each lottery round there are $\boldsymbol{p\ tkt_{rate} draw_{len}}$ winning tickets, each with value $\boldsymbol{\beta}$ coins, then the payment escrow balance is $\boldsymbol{\beta\ p\ tkt_{rate} draw_{len}}$
- Track tickets in the system based on their sequence numbers.
- Miners control escrows in the system.
- Each escrow must identify a set of beneficiary merchants.
- A customer can create an escrow that is sufficient to pay merchants for days.

Lottery Ticket Issuance

- Each ticket is a simple structure consist of:

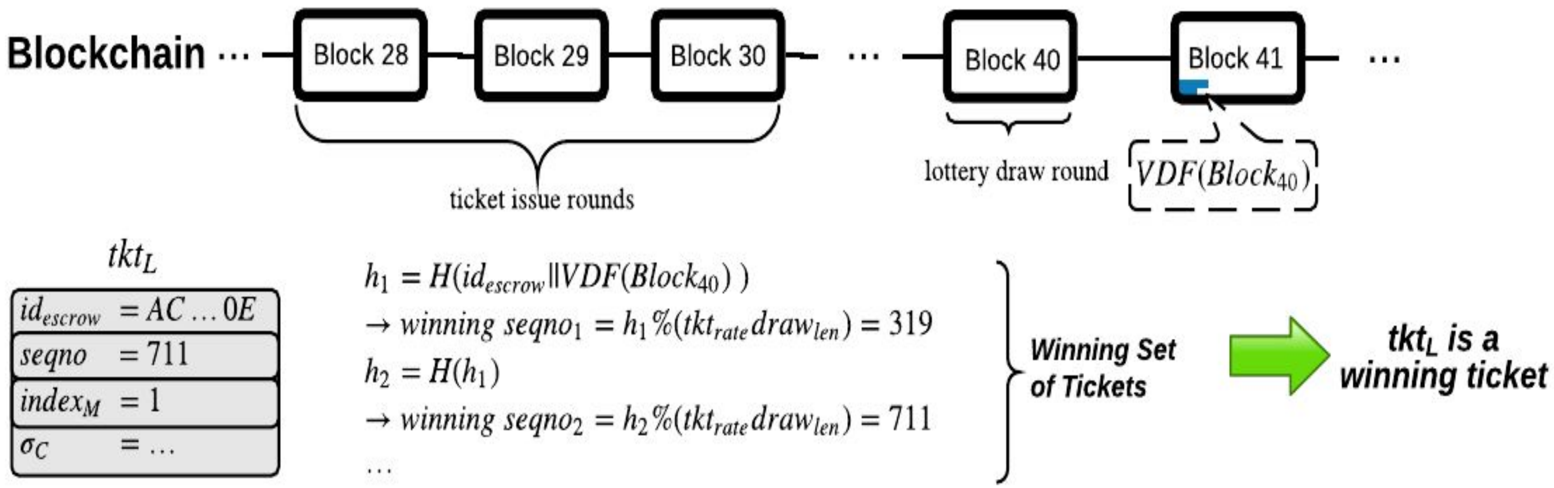
$$tkt_L = id_{esc}||index_M||seqno||\sigma_C$$

- Ticket issuance must follow a ticket issuing schedule.



The lottery Protocol

- Lightweight, non-interactive, and supports exact win rate.
  - Based on the blockchain view and requires only secure hashing.



- Merchants claim their winning tickets through the miners within the ticket redemption period.

Proof-of-cheating Processing

- Any party can issue a proof-of-cheating against the customer if it detects:
  - Duplicate ticket issuance.
  - Issuing more tickets with out-of-range sequence numbers.

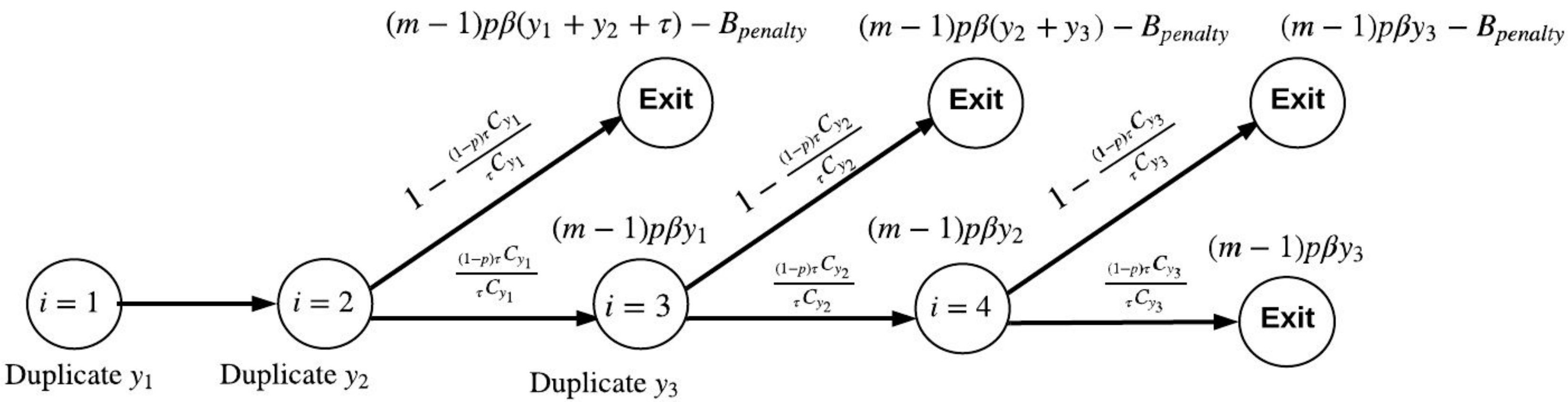
- The miners burn the customer's penalty deposit.
  - This deposit must be large enough to make cheating unprofitable.
  - Its lower bound is derived using a game theoretic analysis of MicroCash setup.

Penalty Deposit I

- Equals at least the additional utility gain a malicious customer obtains over an honest.
- Intuitively, it is the expected amount of payments a customer would pay for (*m*-1) merchants (at max ticket issuance rate) during the cheating detection period.
  - A duplicated ticket is detected after it wins the lottery and is claimed by the marchants.
  - Thus, the cheating detection period covers the lottery period and the ticket redemption period.

Penalty Deposit II

Its lower bound is derived using a game theoretic analysis that models the system as a repeated game and tracks its evolution over time.



$$\mathbb{E}_k[u(\widehat{C})] = \left(1-\frac{(1-p)\tau C_{y_1}}{\tau C_{y_1}}\right)\left((m-1)p\beta\sum_{i=1}^{d} y_i + (m-1)p\beta r\tau - B_{penalty}\right) +$$
$$\left(\frac{(1-p)\tau C_{y_1}}{\tau C_{y_1}}\right)\left((m-1)p\beta y_1 + \mathbb{E}_{k-1}[u(\widehat{C})]\right)$$

Penalty Deposit III

But $\mathbb{E}_{k-1}[u(\widehat{C})] \leq 0$ and $\mathbb{E}_k[u(\widehat{C})] \leq 0$, hence:

$$B_{penalty}(y_1, \ldots, y_d) > (m-1)p\beta\left(\frac{y_1}{1-\frac{(1-p)\tau C_{y_1}}{\tau C_{y_1}}} + \sum_{i=2}^{d} y_i + r\tau \right)$$

The above is maximized when $y_i = (1-p)\tau$ for $i \in \{1, \ldots, d\}$, thus:

$$B_{penalty} > (m-1)p\beta tkt_{rate} draw_{len}\left(\frac{1-p}{1-\rho^{-1}} + draw_{len}\left((1-p)(d_{draw}-1) + d_{redeem}\right)\right)$$

MicroCash Security Properties

- Prevents **escrow overdraft**.
  - Front running attacks are not possible.
  - Ticket tracking prevent issuing more tickets than what can be covered.
- Prevents **escrow-withholding**.
  - An escrow will be refunded once all tickets expire.
- Prevents **manipulating the lottery** outcome.
  - Achieved by the use of VDFs and ticket issuing schedule.
- Addresses **duplicated ticket issuance**.
  - Using detect-and-punish approach.

MicroCash Efficiency - MicroBenchmarks I

- **Ticket processing rate (**ticket / sec**):**

Scheme	ECDSA (secp256k1)	ECDSA (P-256)	EdDSA (Ed25519)
MICROPAY			
Customer	1,859	32,471	26,238
Merchant	1,328	2,399	2,561
Miner	1,340	2,448	2,617
MicroCash			
Customer	1,868	33,006	26,749
Merchant	2,249	10,505	8,473
Miner	2,241	10,345	8,368

Merchants and miners in MicroCash are **1.7x, 4.2x, and 3.2x** faster than in MICROPAY (for the three digital signature schemes shown above).

MicroCash Efficiency - MicroBenchmarks II

- **Bandwidth cost (in terms of ticket size):**
  - From customer to merchant; 274 bytes (MICROPAY), 110 byte (MicroCash, around **60% reduction**).
  - From merchant to miner; 355 byte (MICROPAY), 110 bytes (MicroCash, around **70% reduction**).
- **Number of escrows**:
  - MICROPAY needs **60, 1019, and 653 escrows** to support the rates reported previously.
  - MicroCash needs only ***one escrow***.

In Real World Applications - Online Gaming

Metric	Bitcoin	MICROPAY	MicroCash
Winning tickets / sec	N/A	0.000167	0.000167
Escrows / sec	N/A	0.000552	0.000386
Transactions /sec	16.67	0.000719	0.000552
Transaction fees / round	$680	$0.029341	$0.022541
Bandwidth between customers and miners	3,333 bps	1.105 bps	1.009 bps
Bandwidth between customers and merchants	N/A	36,533 bps	14,667 bps
Bandwidth between merchants and miners	N/A	0.807 bps	0.523 bps
Delta blockchain size / round	2.38 MB	0.000137 MB	0.00011 MB

- **Bitcoin:** Average transaction fee is $0.068, and average transaction size is 250 bytes.
- **Minecraft:** 125 servers, each serving 8 players. Cost is $12 per 8 players per month.
  - With 2% overhead percentage, p = 0.00001
  - Each player pay one ticket per minute.

In Real World Applications - P2P CDNs

Metric	**Bitcoin**	**MICROPAY**	**MicroCash**
Winning tickets / sec	N/A	0.001964	0.001964
Escrows / sec	N/A	0.001976	0.000012
Transactions /sec	128	0.00394	0.001976
Transaction fees / round	$5,222	$0.160769	$0.08062
Bandwidth between customers and miners	256,000 bps	3.95 bps	0.165 bps
Bandwidth between customers and merchants	N/A	280,576 bps	112,640 bps
Bandwidth between merchants and miners	N/A	9.508 bps	6.16 bps
Delta blockchain size / round	18.31 MB	0.000963 MB	0.000452 MB

- **CDN:** one publisher serving 1 Gpb, cost is $0.01, each cache gets a ticket per 1 MB it serves..
    - With 2% overhead percentage, p = 0.000023
    - Issues 128 tickets per second

Conclusions

- Micropayments have a large number of potential applications.
  - Cryptocurrencies provided a template to recast centralized probabilistic micropayments into distributed ones.
- Microcash is the first distributed probabilistic micropayment scheme that supports concurrent micropayments with exact win lottery protocol.
- It is also efficient, its non-interactive lottery requires only one round of communication and relies only on secure hashing.
- Results confirm its variability to be used in large-scale distributed systems.

Thank You!

Questions?

References

[Rivest, 1997] Ronald Rivest.1997.Electronic lottery tickets as micropayments. In International Conference on Financial Cryptography. Springer, 307–314.

[Wheeler, 1996] David Wheeler. 1996. Transactions using bets. In International Workshop on Security Protocols. Springer, 89–92.

[Pass et al., 2015] Pass, Rafael. "Micropayments for decentralized currencies." In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pp. 207-218. ACM, 2015.

[Chiesa et al., 2017] Chiesa, Alessandro, Matthew Green, Jingcheng Liu, Peihan Miao, Ian Miers, and Pratyush Mishra. "Decentralized Anonymous Micropayments." In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 609-642. Springer, Cham, 2017.