Building Secure Distributed Services and Resource Markets

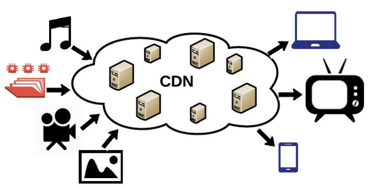
Ghada Almashaqbeh
CacheCash and NuCypher

Traditional Service Systems

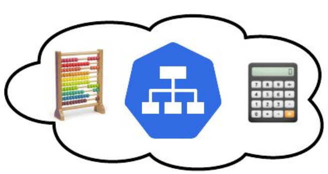
Central Management

Services

File Storage



Content Distribution



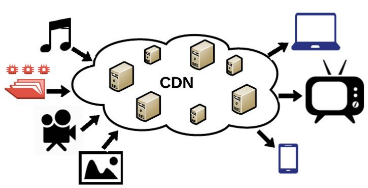
Computing

Payments

Traditional Service Systems

Central Management

Services



Content Distribution

Online Content Distribution

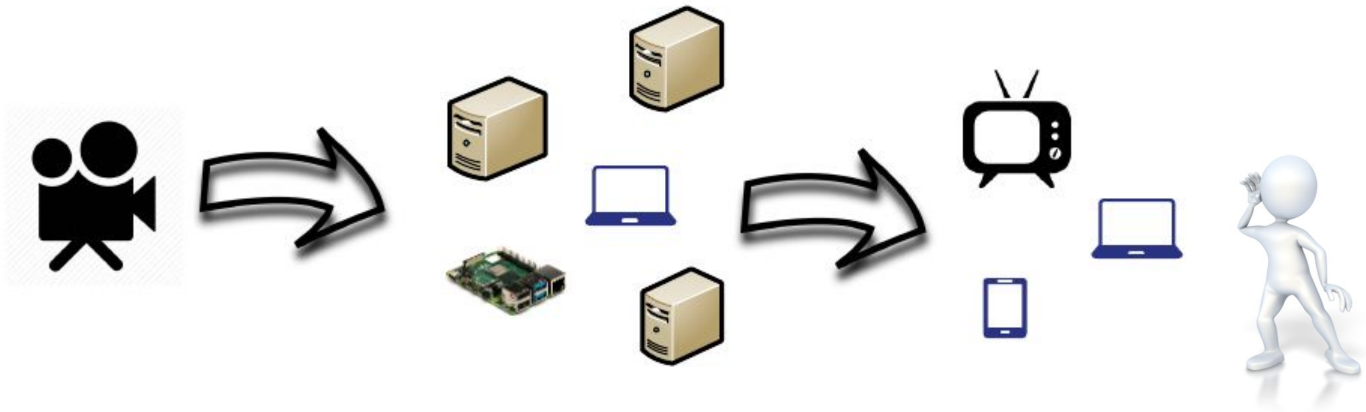
- Dramatic growth over the past decade.
- Usually, infrastructure-based CDNs are used to distribute the load.
  - Through CDN providers, e.g., Akamai.

- **Drawbacks:**
  - Costly and complex business relationships.
  - Over-provisioning bandwidth needs.
  - Issues related to reachability, visibility, flexibility, etc.

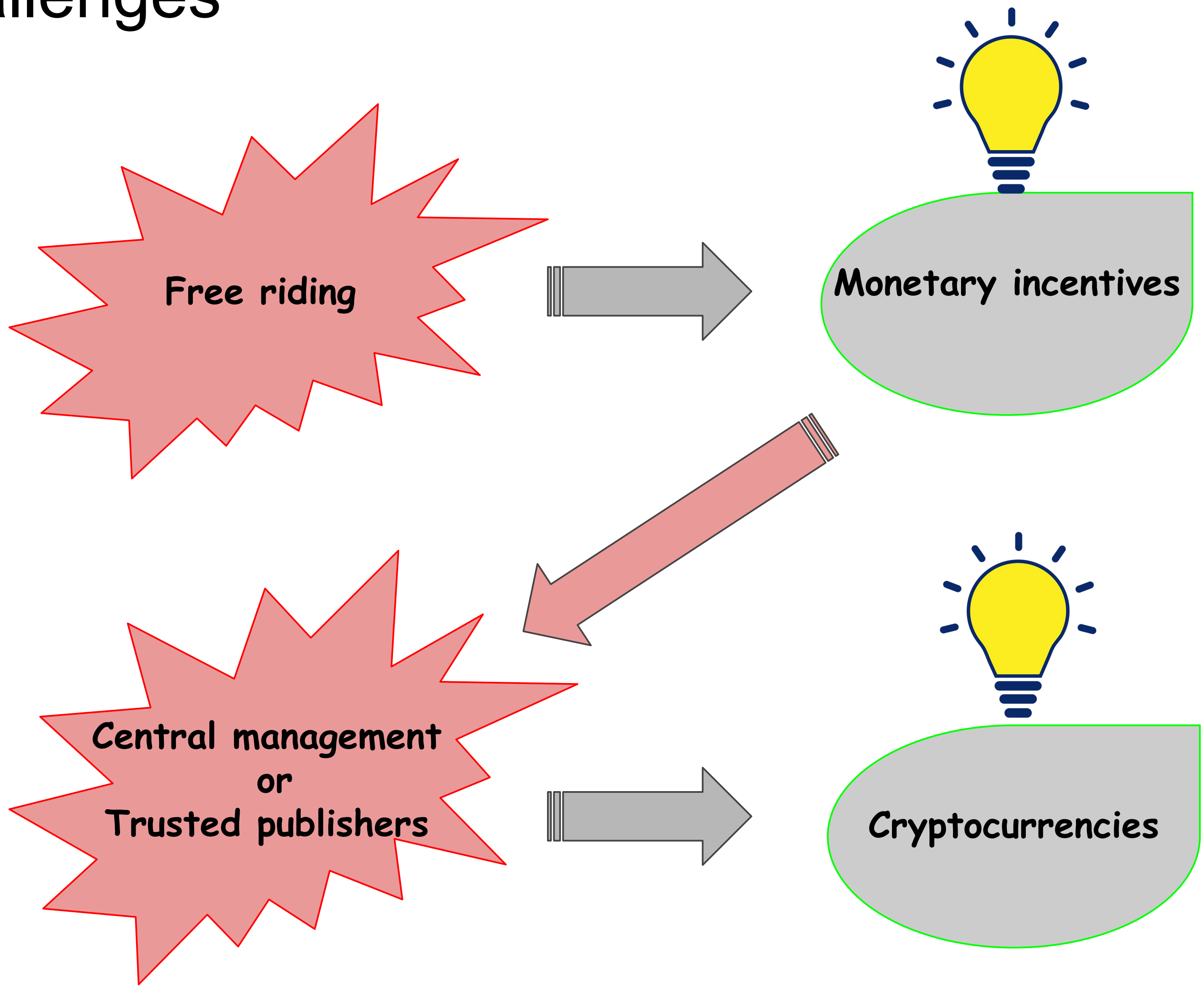
Decentralized CDNs

- Utilize P2P data transfers to build dynamic CDNs.

- **Advantages:**
  - Flexible CDN service.
  - Easier to scale with demand.
  - Extended reachability and lower latency.
  - Democratized and transparent service.

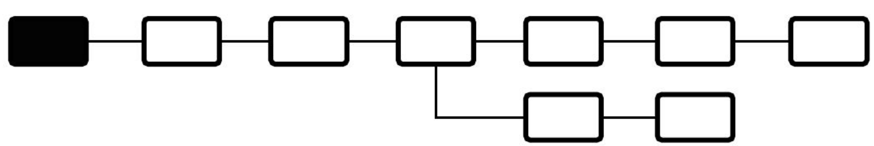


Challenges

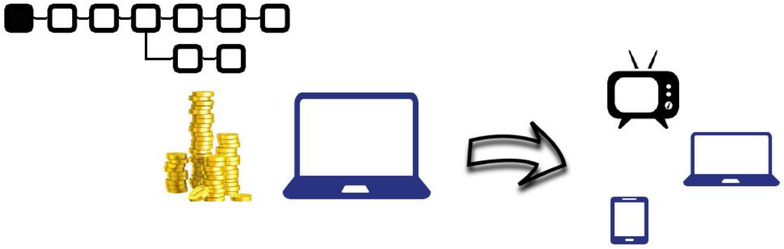

Free riding
Monetary incentives
Central management
or
Trusted publishers
Cryptocurrencies

Cryptocurrencies and Blockchain Technology

- An emerging economic mechanism that received a huge interest.
- Early systems focused on providing a currency exchange medium.
    - Distributed, publicly verifiable, open to anyone.
- Newer systems provide a service on top of this medium.
    - Create ***distributed resource markets***.
    - E.g., Golem, Filecoin, Livepeer.



Build a distributed bandwidth market!

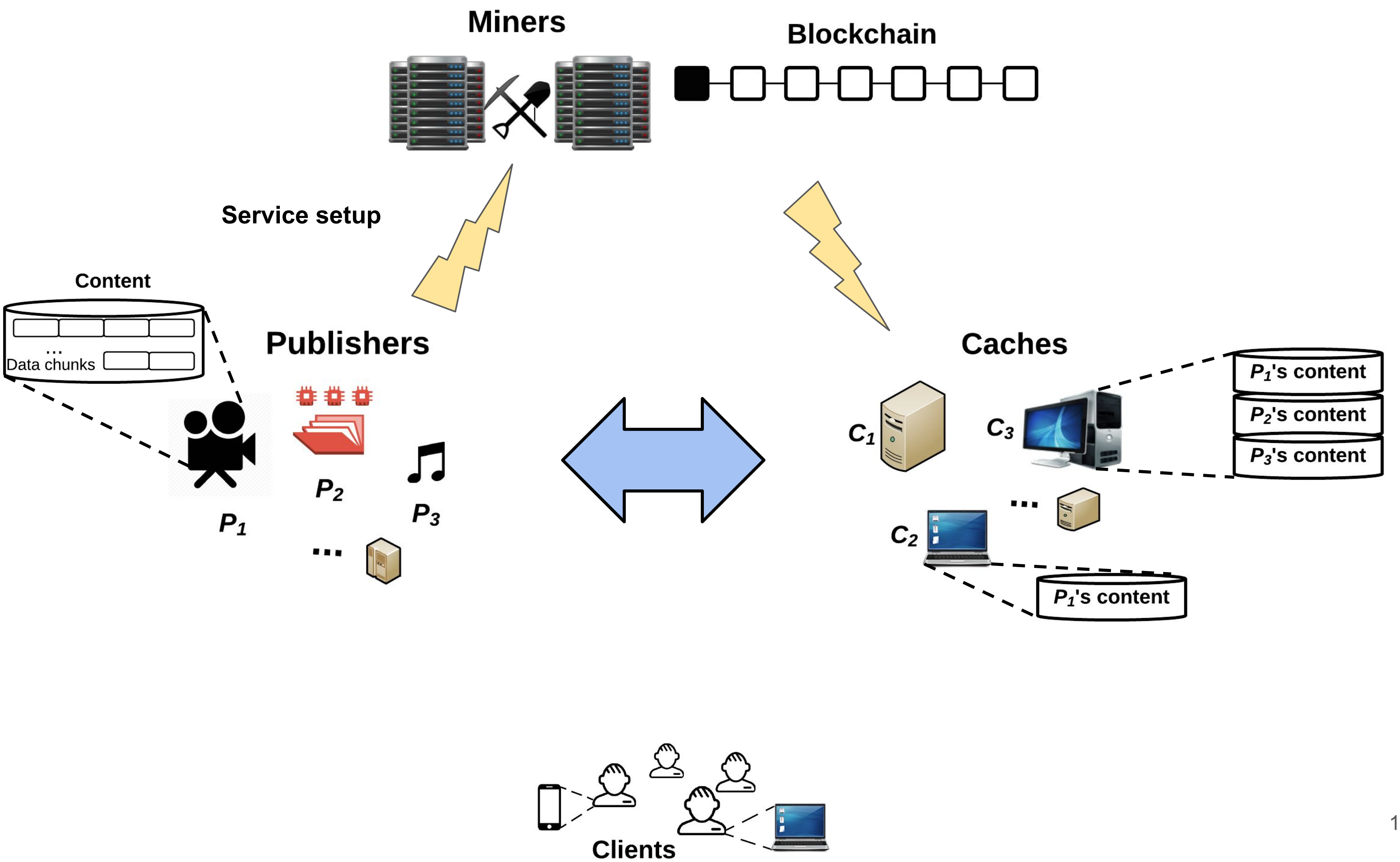


cachecash

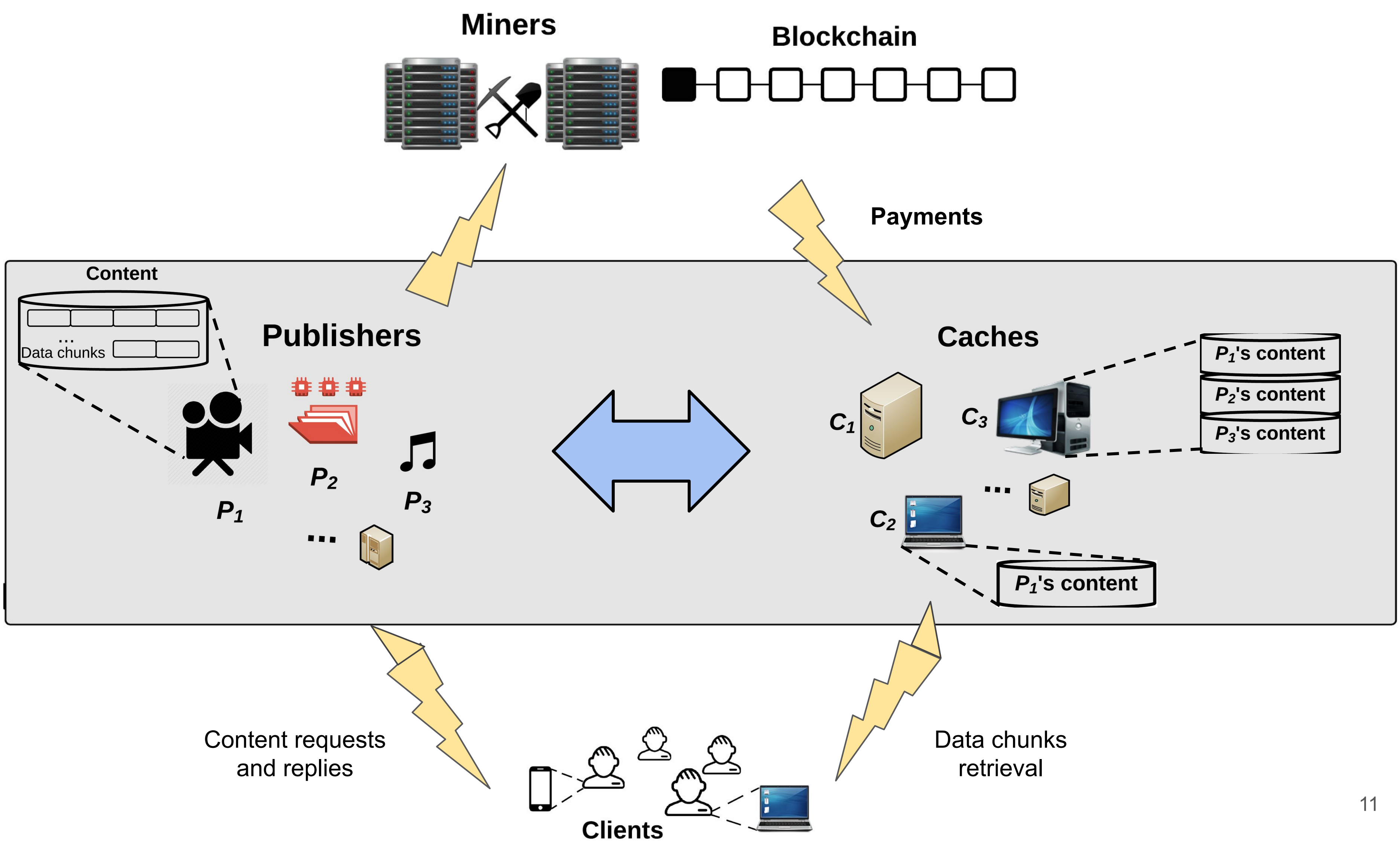
CacheCash

- A decentralized CDN service powered by a cryptocurrency.
  - A distributed, trustless bandwidth market.
    - Open access.
    - A novel service-payment exchange protocol.
    - A unique service pricing mechanism.
  - Secure.
    - Cryptographic and game theoretic security defenses.
  - Efficient.
    - Several performance optimization techniques.

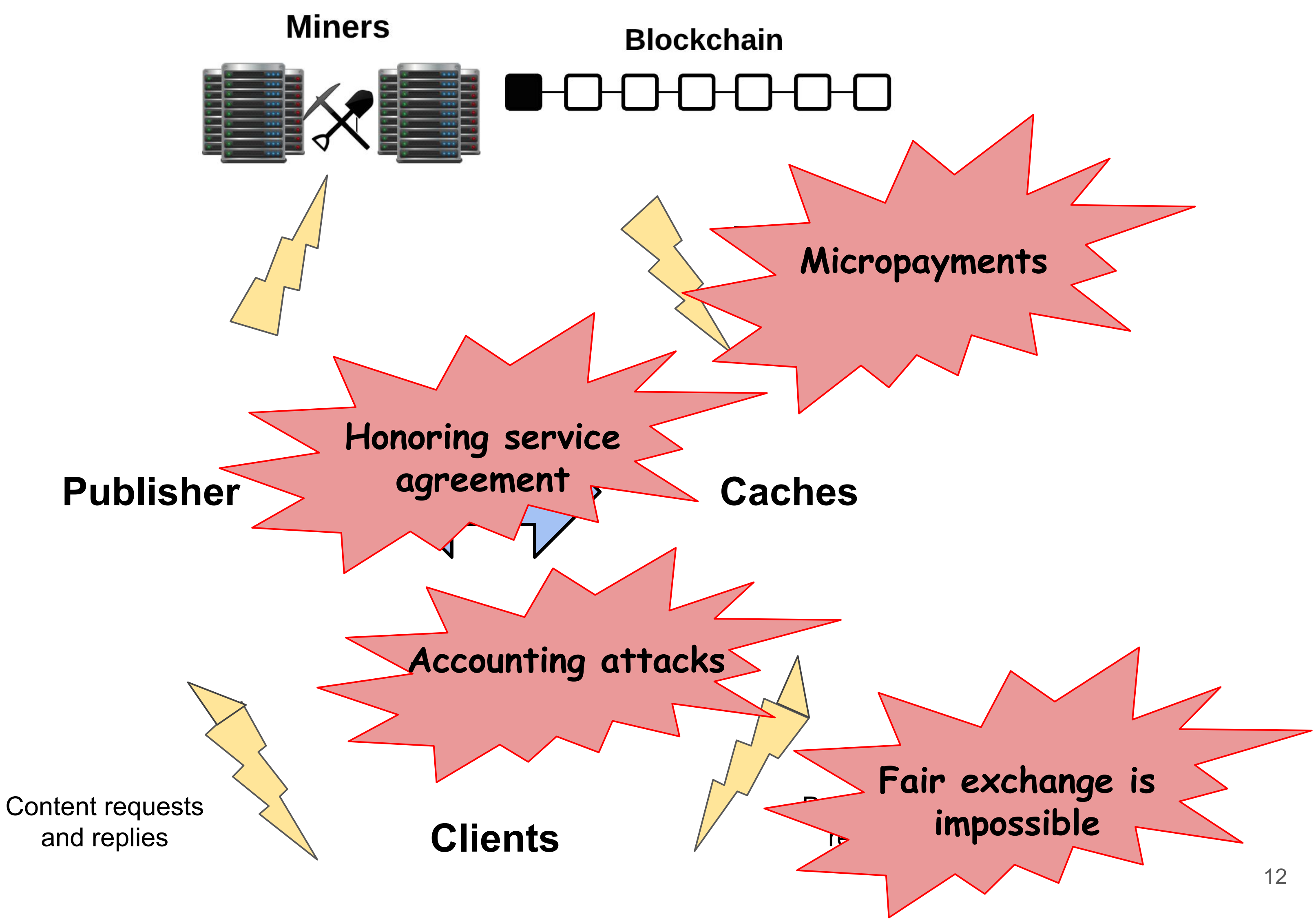
CacheCash Pictorially



CacheCash Pictorially



Challenges


Miners
Blockchain
Micropayments
Honoring service agreement
Publisher
Caches
Accounting attacks
Content requests and replies
Clients
Fair exchange is impossible

Micropayments

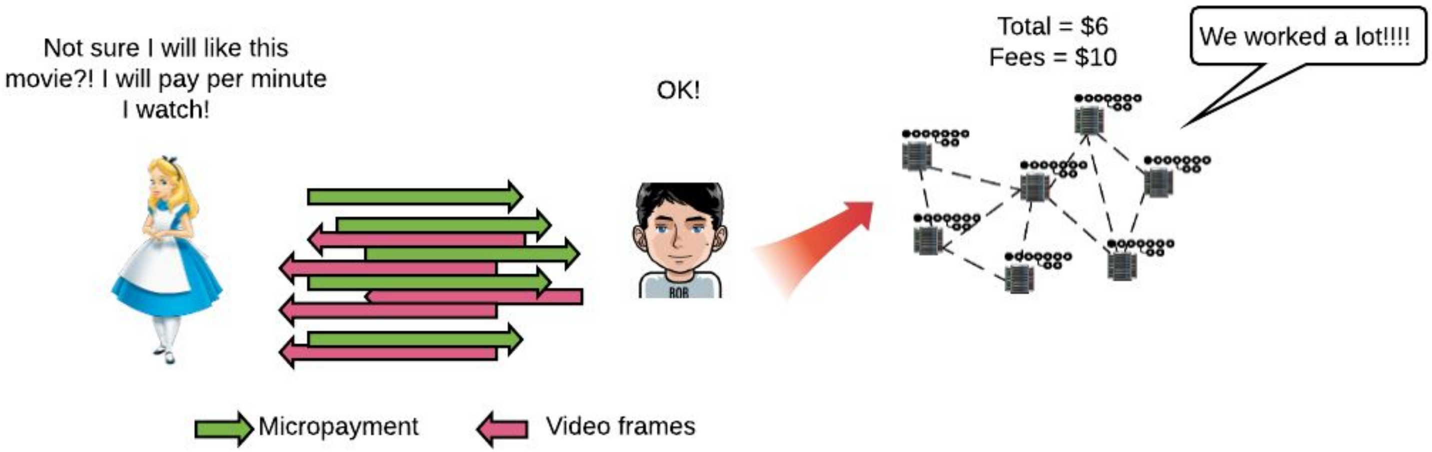
MicroCash:
Practical Concurrent Processing of Micropayments

[Financial Crypto'20]

"Micropayments are back, at least in theory, thanks to P2P" *

Micropayments

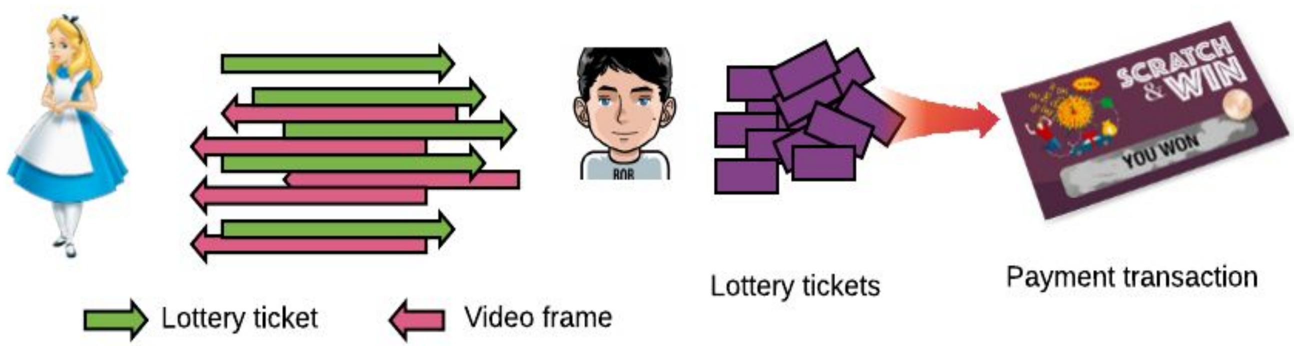
- Several potential applications.
  - Ad-free web surfing, online gaming, rewards in P2P systems, etc.
- ***Advantages;*** flexible and reduce financial risks.
- ***Drawbacks***; high transaction fees and large system load.



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

Probabilistic Micropayments

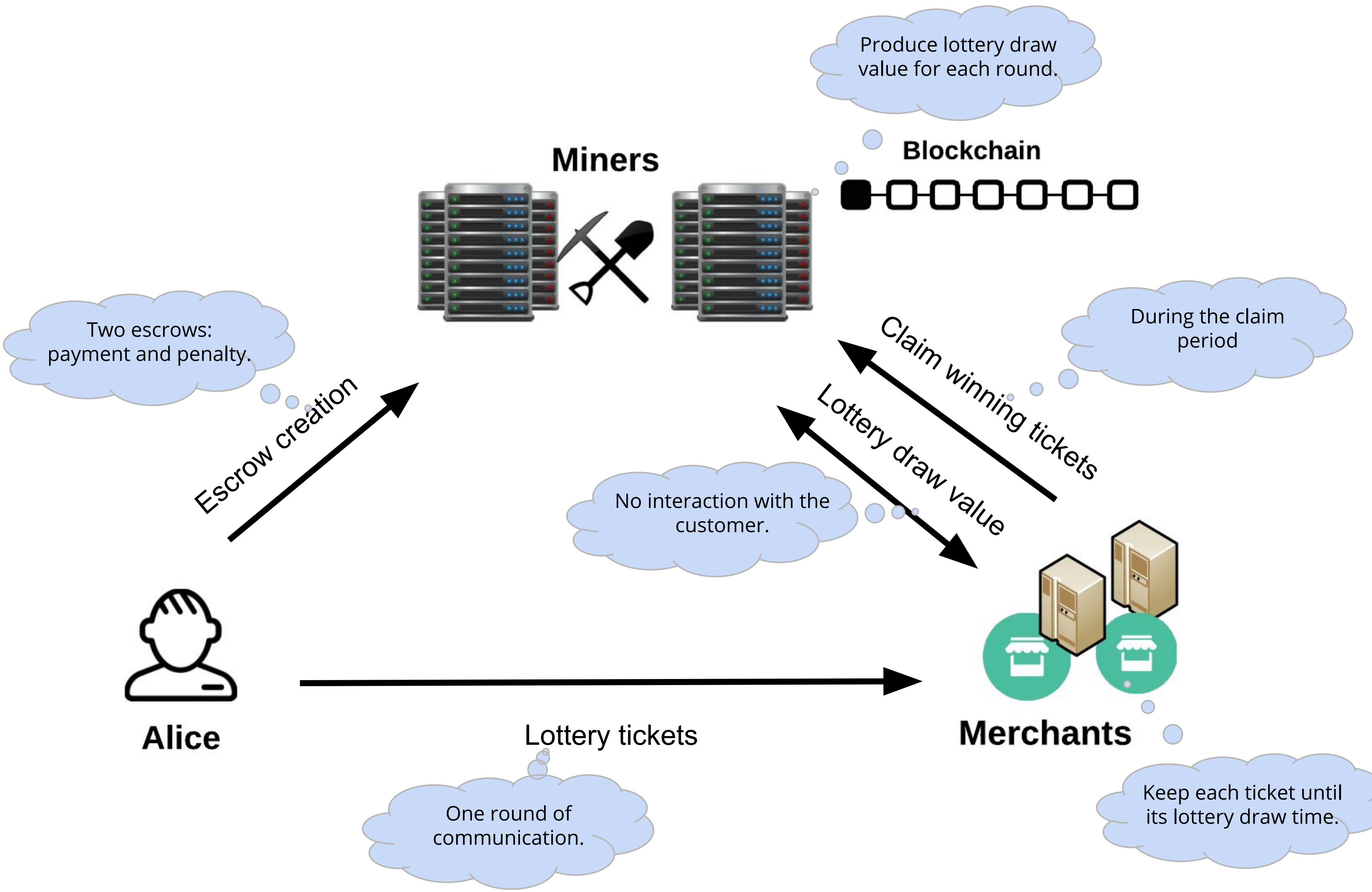
- A solution to aggregate tiny payments.
- Dated back to Wheeler [W96] and Rivest [R97].



- Cryptocurrencies are utilized to achieve decentralization.
- Prior work: **MICROPAY** [PS15] and **DAM** [CGL+17]
    - Sequential, interactive lottery protocol, computationally-hea

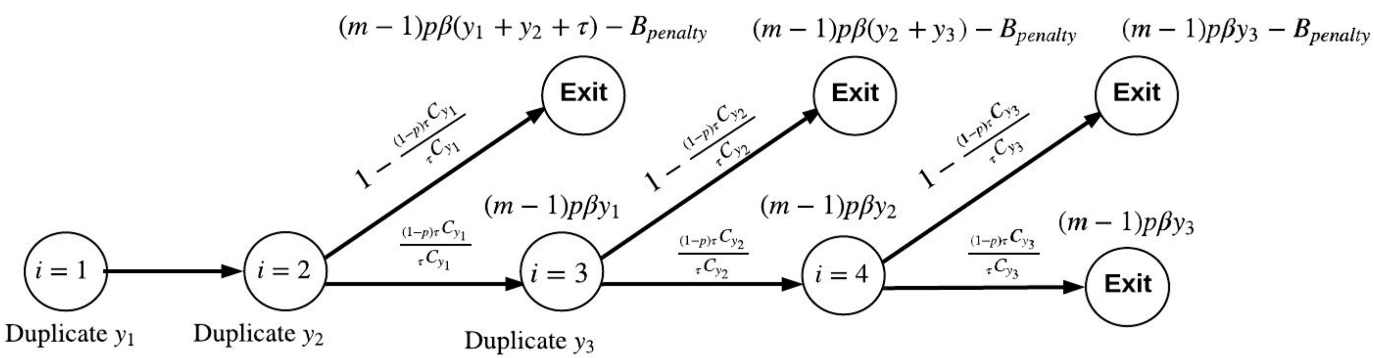
MicroCash

MicroCash in a Nutshell



Penalty Escrow

- Used to defend against ticket duplication.
  - Equals at least the additional utility a malicious customer obtains over an honest.



Theorem. *For the game setup of MicroCash, issuing invalid or duplicated lottery tickets is less profitable in expectation than acting in an honest way if:*

$$B_{penalty} > (m-1)p\beta\tau\left(\frac{1-p}{1-\frac{1}{\tau C_{(1-p)\tau}}} + (1-p)(d-1) + r\right)$$

MicroCash ⇒ Reward Caches

Payments are well deserved?!

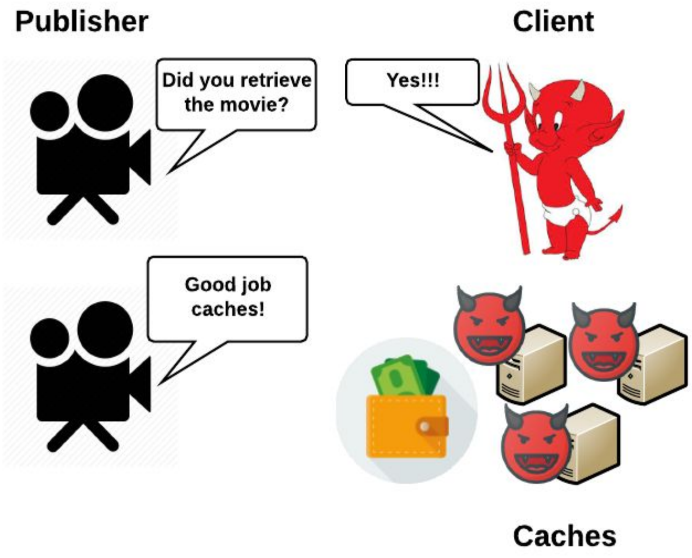
Accounting Attacks

CAPnet:
A Defense Against Cache Accounting Attacks

[IEEE CNS'19]

Background

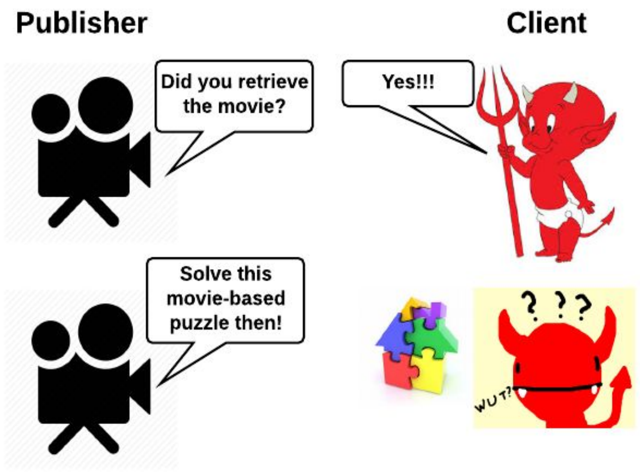
- **Cache accounting attacks**.
  - Allow caches to collect rewards for free.
  - Mislead network resource management.



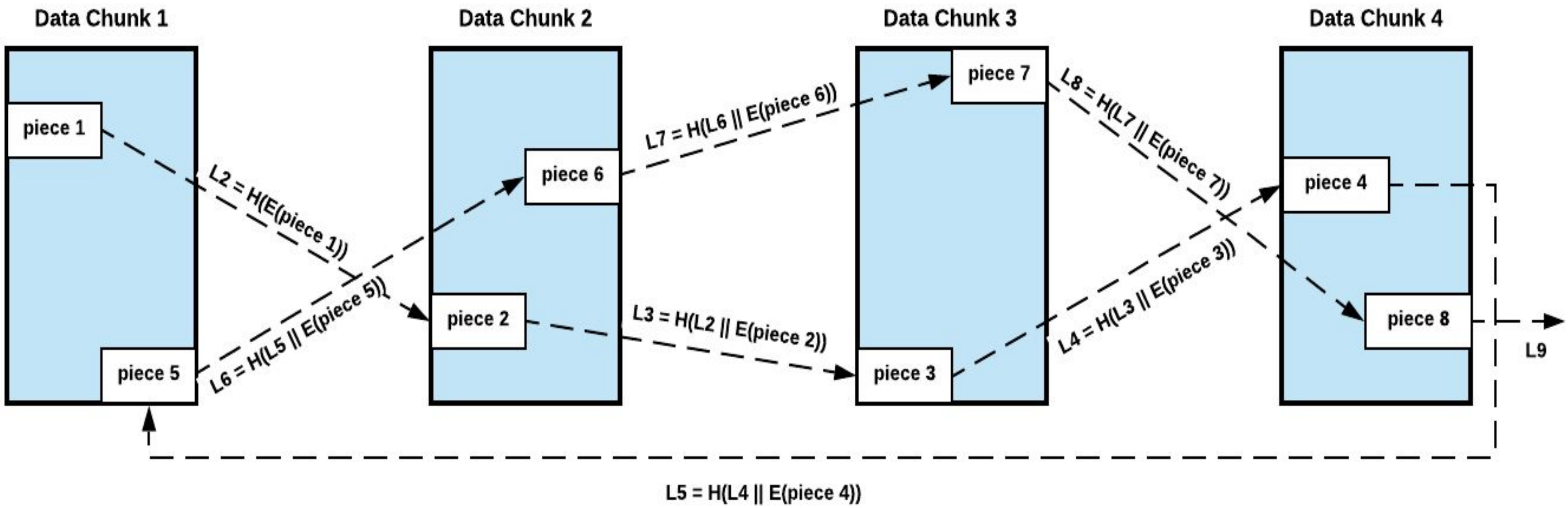
- **Previous solutions:** Do not work in typical P2P networks.
  - Either rely on activity reports from the peers themselves.
  - Or assume the knowledge of peer computational power and link delay.

CAPnet

- Lets untrusted caches join peer-assisted CDNs.
- Introduces a lightweight cryptographic puzzle that ensures content retrieval.
  - Its security is a financial one (in terms of bandwidth consumption).
- Allows a publisher to set the desired tradeoff between security and efficiency.



Data Colocation Puzzle Design



Puzzle challenge = $H(L_9)$ Puzzle solution = L_9

Fair exchange is impossible

Honoring service agreement

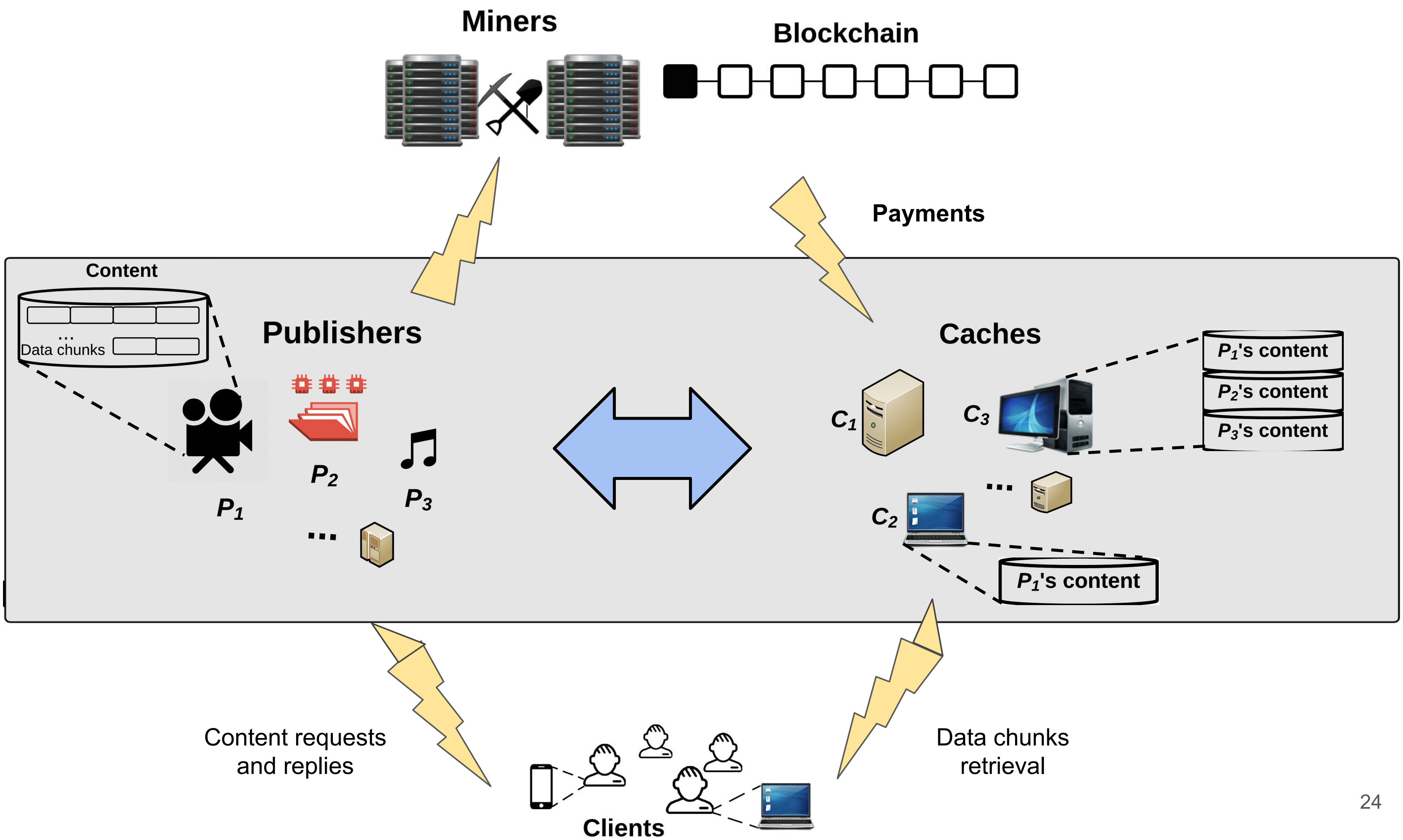
CacheCash

Service-payment exchange protocol

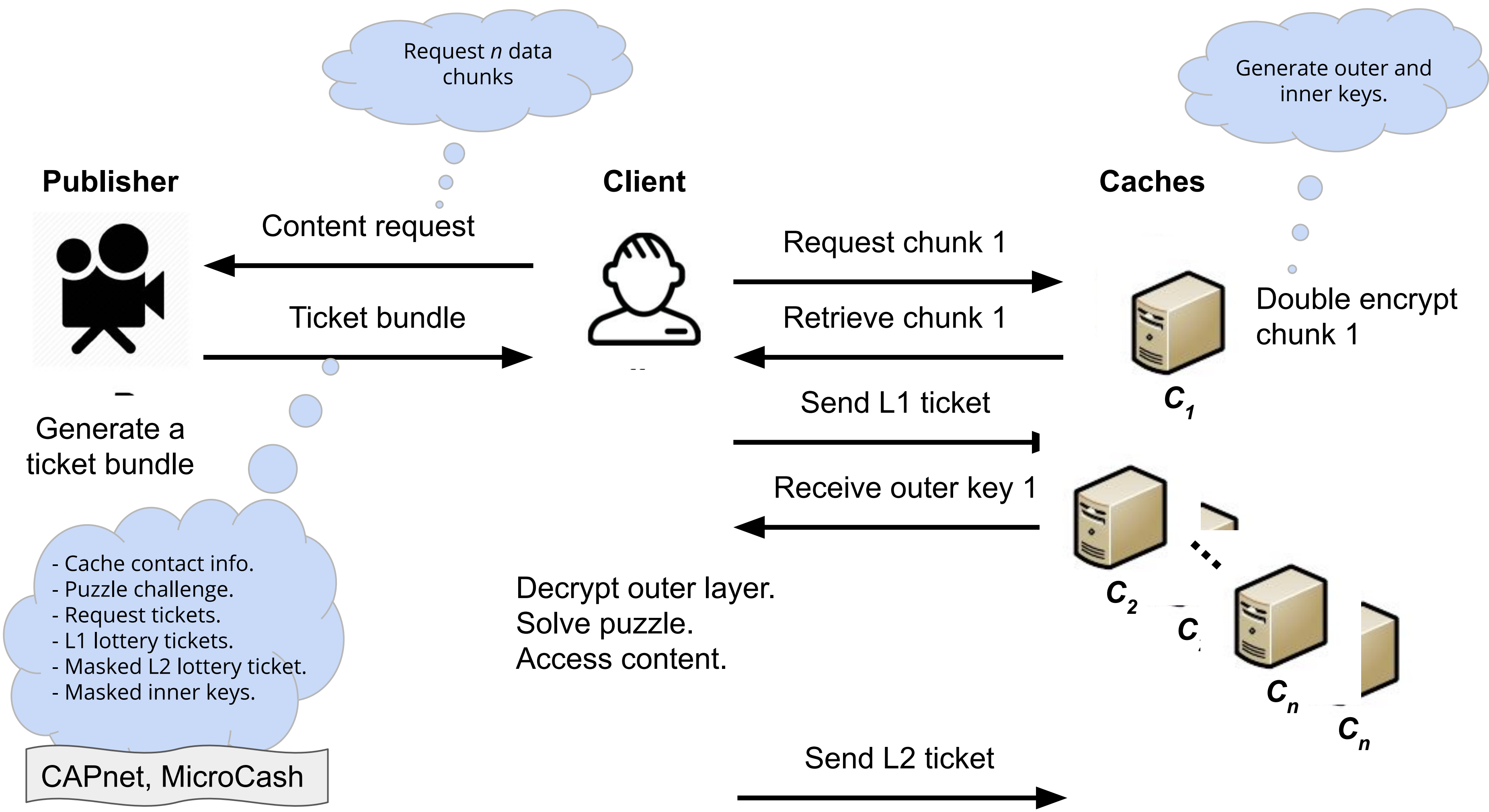
Service pricing

cachecash [Startup founded in 2018]

CacheCash - Recall



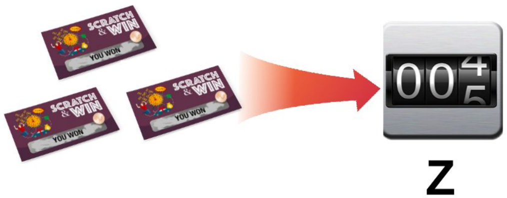
Content Distribution



Payment Processing

- Caches claim winning lottery tickets, same as in MicroCash.
- However, computing the payment value is different.

Winning L2 tickets



Winning L1 tickets

$$f(N_i, Z) = \alpha\sqrt{N_i Z}$$

- This also requires deriving new bounds for the payment and penalty escrows.

Payment Escrow

- Guarantees that a publisher can pay all winning tickets tied to the escrow.
  - Accounts for the worst case by quantifying over the most expensive service price among all beneficiary caches.

Theorem. *For the payment configuration of CacheCash, the currency balance needed to cover all winning tickets tied to an escrow is given by:*

$$B_{escrow} = \alpha_{max} p \; draw_{len} \sqrt{n_{max} tkt_{rate1} tkt_{rate2}} \left(\frac{2}{3} \left(\frac{l_{esc}}{draw_{len}} + 1 \right)^{1.5} + 0.5 \sqrt{\frac{l_{esc}}{draw_{len}} + 1} \right)$$

Penalty Escrow

Theorem. *For the game setup of CacheCash, issuing invalid or duplicated lottery tickets is less profitable in expectation than acting in an honest way if:*

$$B_{penalty} > \frac{\alpha_{max} p \tau_2 (N-1)}{1-\rho} \sqrt{\left(\frac{n_{max}(v-d-r)}{N} + 1 \right)} + \alpha_{max} p \tau_2 (N-1) \sum_{i=2}^{d+r} \sqrt{\left(\frac{n_{max}(v-d-r)}{N} + i \right)}$$

Such that:

$$\rho = \begin{cases} \dfrac{\binom{(1-p)\tau_1}{\tau_2}}{\binom{\tau_1}{\tau_2}\binom{\tau_2}{(1-p)\tau_2}} & \text{if } (1-p)\tau_1 \geq \tau_2 \\[2ex] \dfrac{1}{\binom{\tau_1}{(1-p)\tau_2}\binom{\tau_2}{(1-p)\tau_2}} & \text{if } \tau_1 = \tau_2 \end{cases}$$

Done right?!

Security

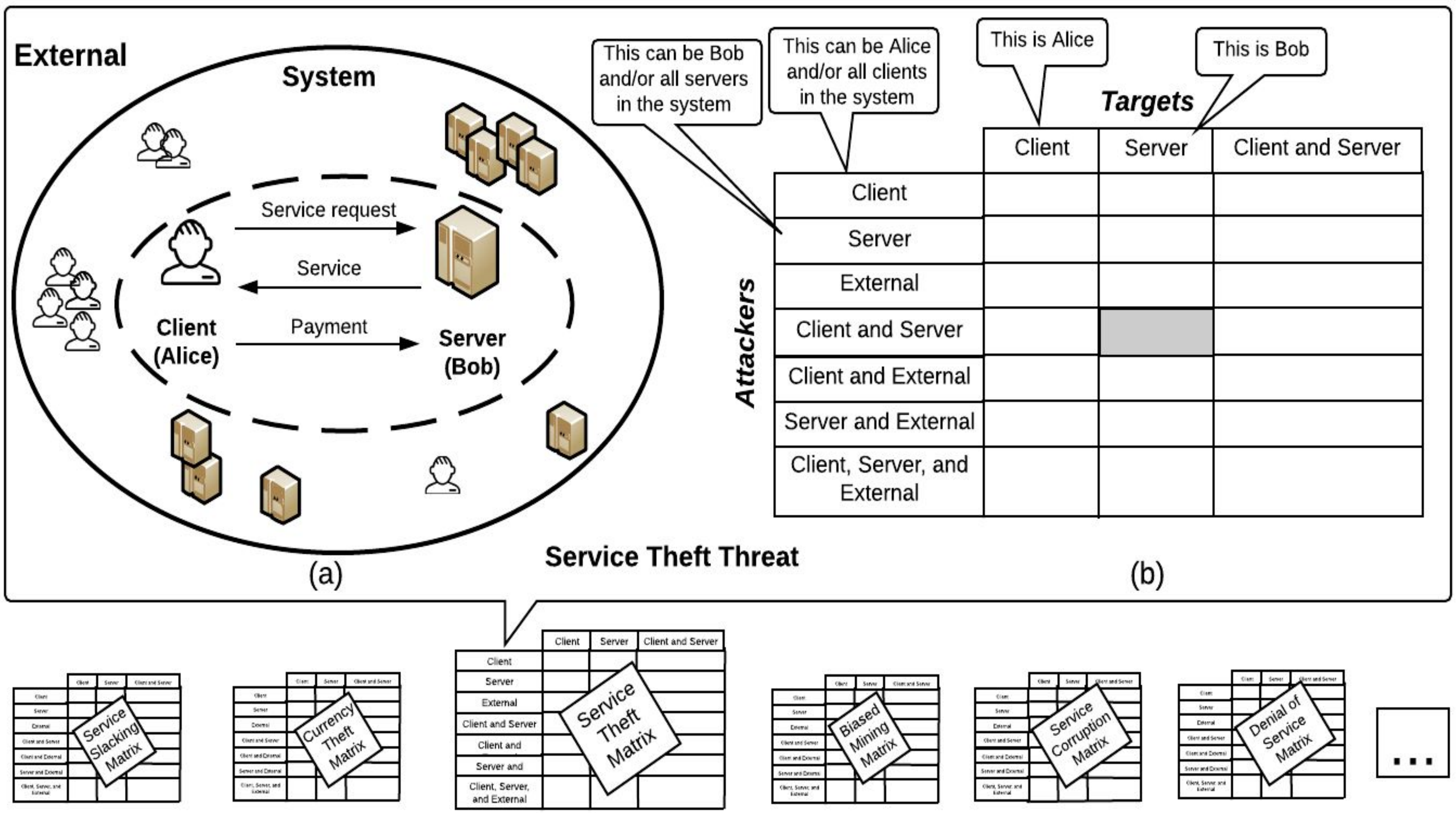
Efficiency

ABC:
A Cryptocurrency-focused Threat Modeling Framework
[CryBlock'19]

What is ABC?

- A systematic threat modeling framework geared toward cryptocurrency-based systems.
    - A qualitative way of analyzing security.
- Helps system designers to consider:
    - Financial motivation of attackers.
    - New asset types in cryptocurrencies.
    - System-specific threat categories.
    - Collusion by using a new tool called a collusion matrix.
        - Also manages the complexity of the threat space.
- Acknowledges that financial incentives can play a major role in other steps in the design process.

A Collusion Matrix Example



CacheCash Security Properties

- ABC was used to build a thorough threat model of CacheCash and its modules.
    - Total **651** threat cases reduced to **32** impactful threat scenarios.
- Addresses **service corruption**.
- Defends against **cache accounting attacks** and other **payment module related threats**.
    - By the security of CAPnet and MicroCash.
- Handles **service theft attacks** and **violating the escrow terms**.
    - Penalty escrow.
    - Novel ***service pricing mechanism*** that incentivizes the honest behavior.

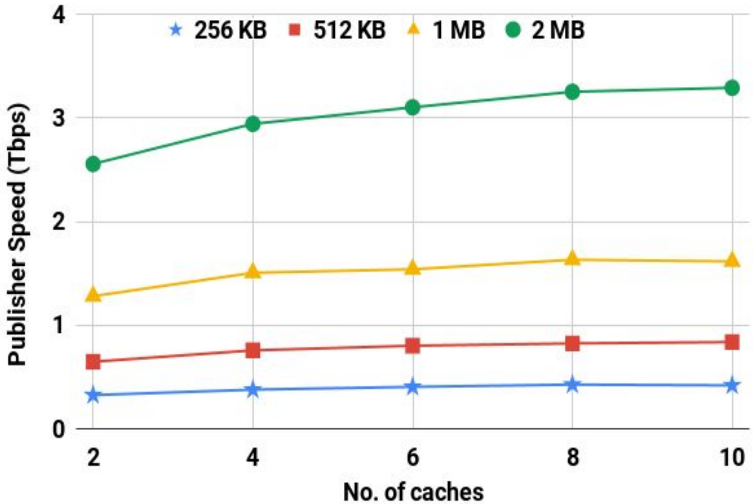
Implementation and Testing

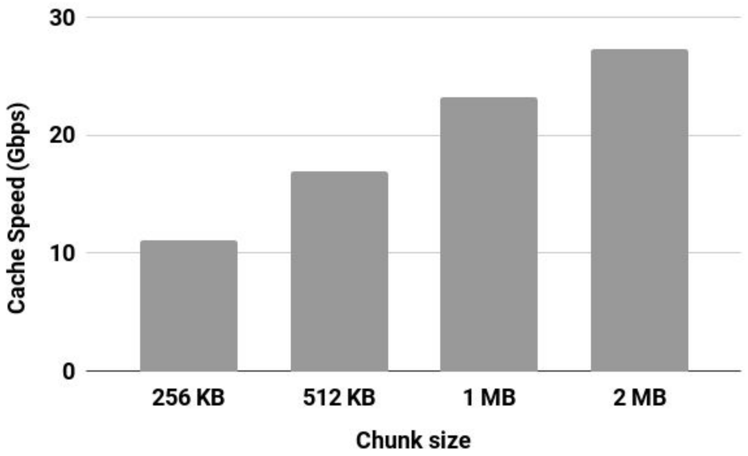
Prototype

Thorough benchmarks

Testnet

CacheCash Efficiency





** Publisher ⇒ serve **315,780** clients watching the same 1080p video concurrently.

** Client ⇒ retrieve content at a rate of **122 Mbps** (watch **24** 1080p videos concurrently).

** Bandwidth overhead is less than **0.1%.**

Current and Future Work

My Contributions

Secure Distributed Systems

ABC (CryBlock'19) *Best Paper Award*
CAPnet (IEEE CNS'19)
MicroCash (Financial Cryptography'20)
CacheCash *Startup founded in 2018*

Privacy Preserving Computing

Private genome testing (BMC'15)
Private compiler extension (TOPS'17)
Gage MPC (Under review)

Basing Cryptography on Physical Assumptions

Basing cryptography on biological polymers (in progress)

Wireless Networks

WSNs, WMNs, WBANs.
(SensorComm'07, WPC'09, ICCSII'12, MedSys'14, BodyNets'14, WINET'14, Sensors'16, …)

Future Work Directions

Heterogeneous Environments

Hybrid systems.
Various capability classes.
Various security/privacy requirements.

Blockchain-based Systems

Explore layer 1 (useful mining, mining pools, …)

Cryptography/Security and Other Fields

Unconventional adversarial models.
Unconventional hardness assumptions.


THANK YOU