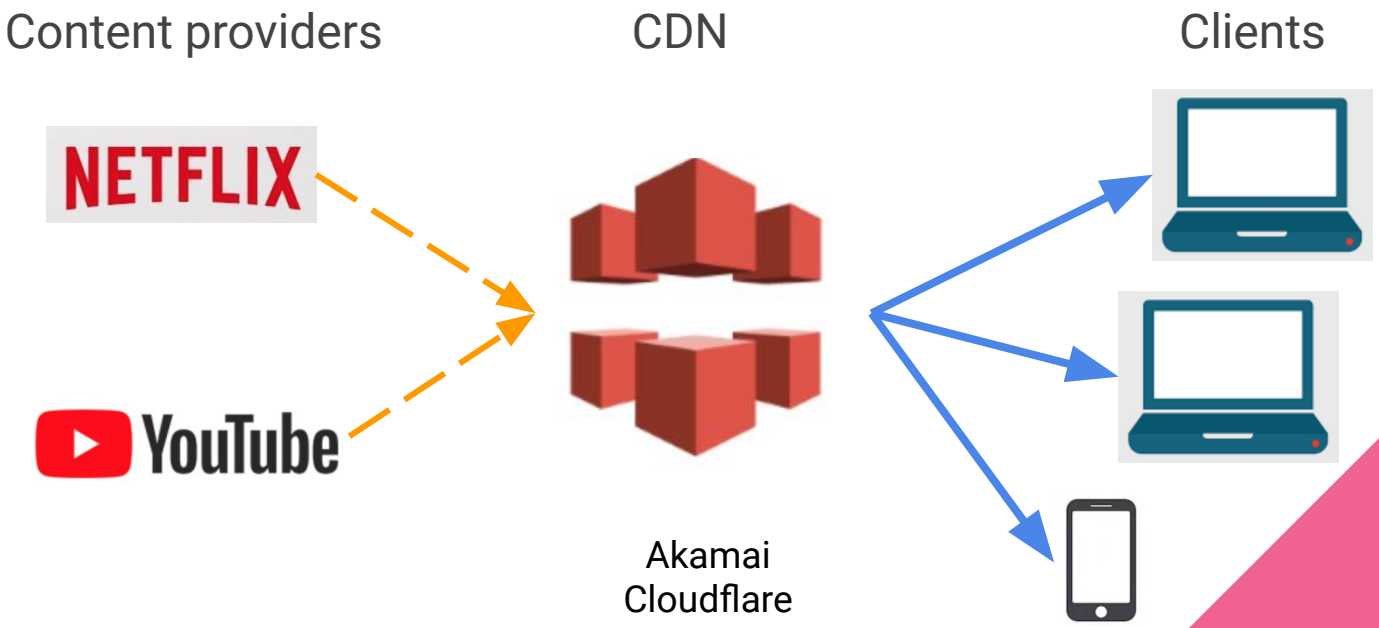
Secure Incentivization for Decentralized Content Delivery

Prateesh Goyal[1], Ravi Netravali[2], Mohammad Alizadeh[1], Hari Balakrishnan[1]

MIT CSAIL[1], UCLA[2]

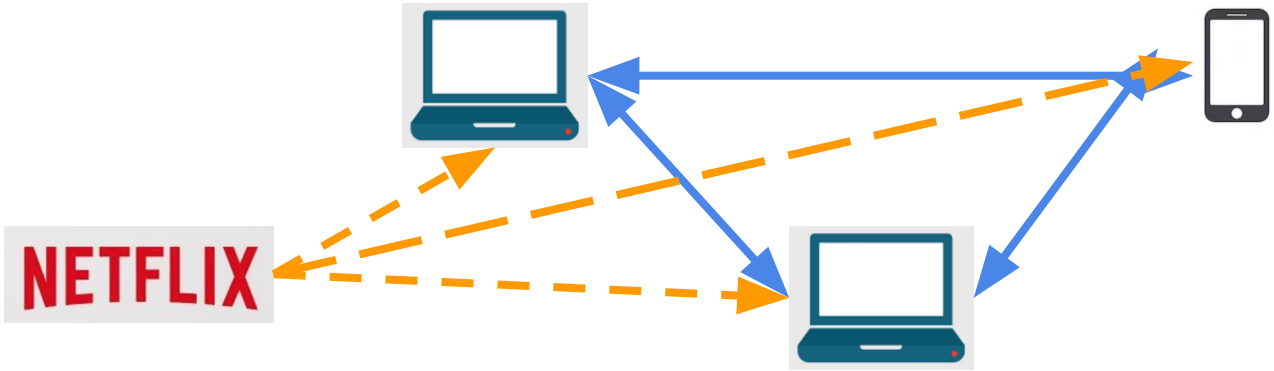
Reducing price for content delivery

- CDNs - Good QoS but expensive for content providers



Reducing price for content delivery

- P2P content delivery (e.g., Akamai Netsession, Peer5)
  - Use unused bandwidth and storage resource at peers
  - Cheaper but insufficient participation by peers [1]

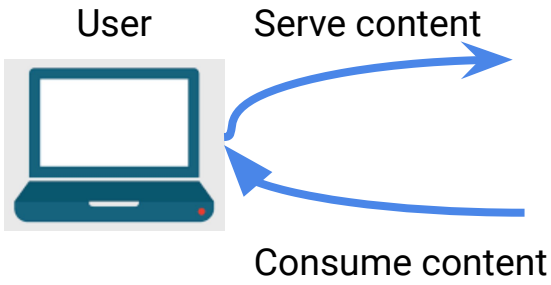


[1] - N. Anjum, et al. Survey on peer-assisted content delivery

Increasing adoption by peers

Tit-for-tat incentivization (e.g., bit torrent, dandelion)

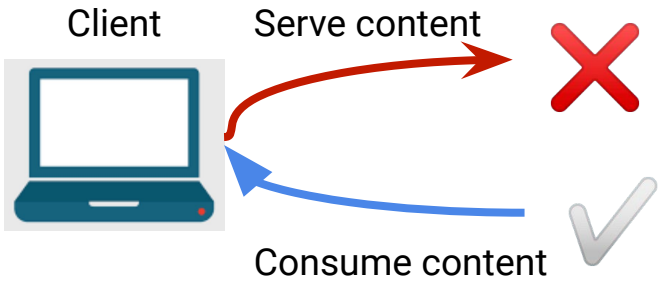
- To consume content, clients must serve content (act as peers)
- Doesn't work for content delivery



Increasing adoption by peers

Tit-for-tat incentivization (e.g., bit torrent, dandelion)

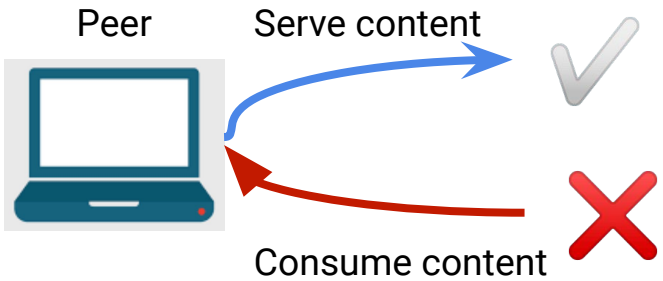
- To consume content, clients must serve content (act as peers)
- Doesn't work for content delivery
  - Restricts clients who do not wish to contribute resources



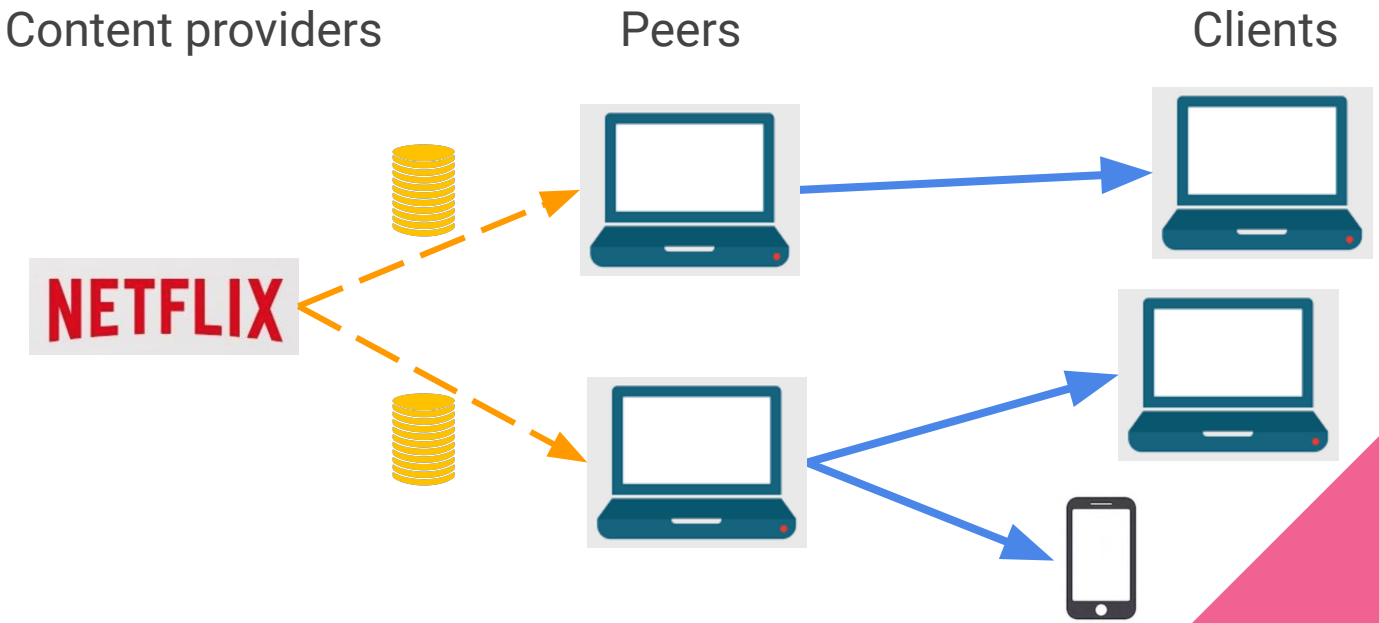
Increasing adoption by peers

Tit-for-tat incentivization (e.g., bit torrent, dandelion)

- To consume content, clients must serve content (act as peers)
- Doesn't work for content delivery
  - Restricts clients who do not wish to contribute resources
  - Fails to provide incentivizes to peers who do not wish to consume the content

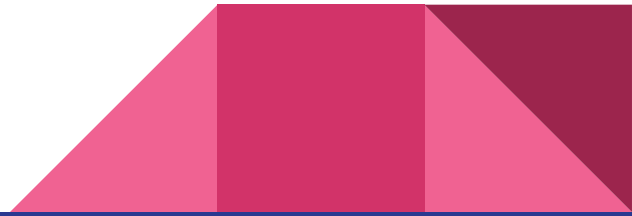


Gringotts: Monetary incentivization for peers



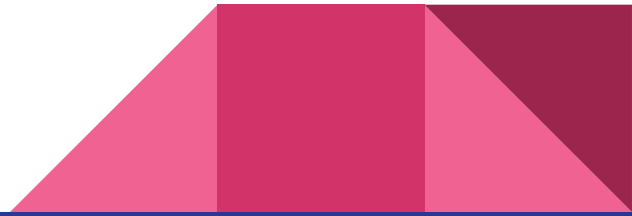
Contributions

- Proof of Delivery (PoD)
  - Peers get paid by content providers for delivering content
  - Secure against financially motivated attacks. E.g., Peer gets paid without delivering content
- Decentralized P2P content delivery
  - No centralized authority (trustless)
  - Peers ensured payment by broadcasting PoD on blockchain
- Consumer Survey
  - Sample size: 876-person
  - 51% respondents would participate



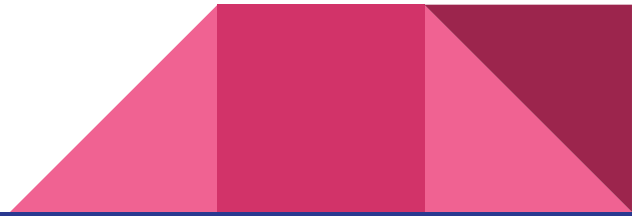
Contributions

- Proof of Delivery (PoD)
  - Peers get paid by content providers for delivering content
  - Secure against financially motivated attacks. E.g., Content provider refuses to pay peer
- Decentralized P2P content delivery
  - No centralized authority (trustless)
  - Peers ensured payment by broadcasting PoD on blockchain
- Consumer Survey
  - Sample size: 876-person
  - 51% respondents would participate

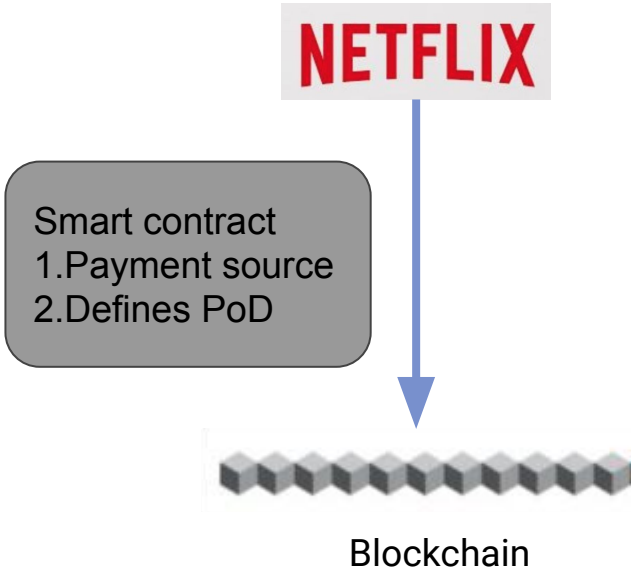


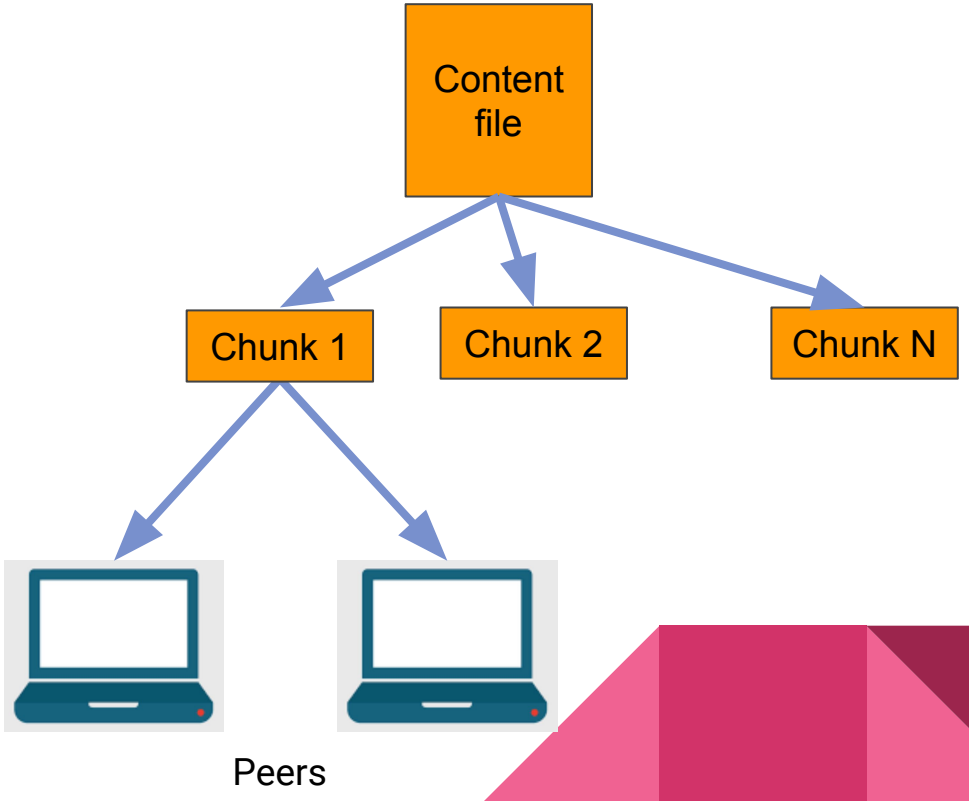
Contributions

- Proof of Delivery (PoD)
  - Peers get paid by content providers for delivering content
  - Secure against financially motivated attacks. E.g., Content provider refuses to pay peer
- Decentralized P2P content delivery
  - No centralized authority (trustless)
  - Peers ensured payment by broadcasting PoD on blockchain
- Consumer Survey
  - Sample size: 876-person
  - 51% respondents would participate

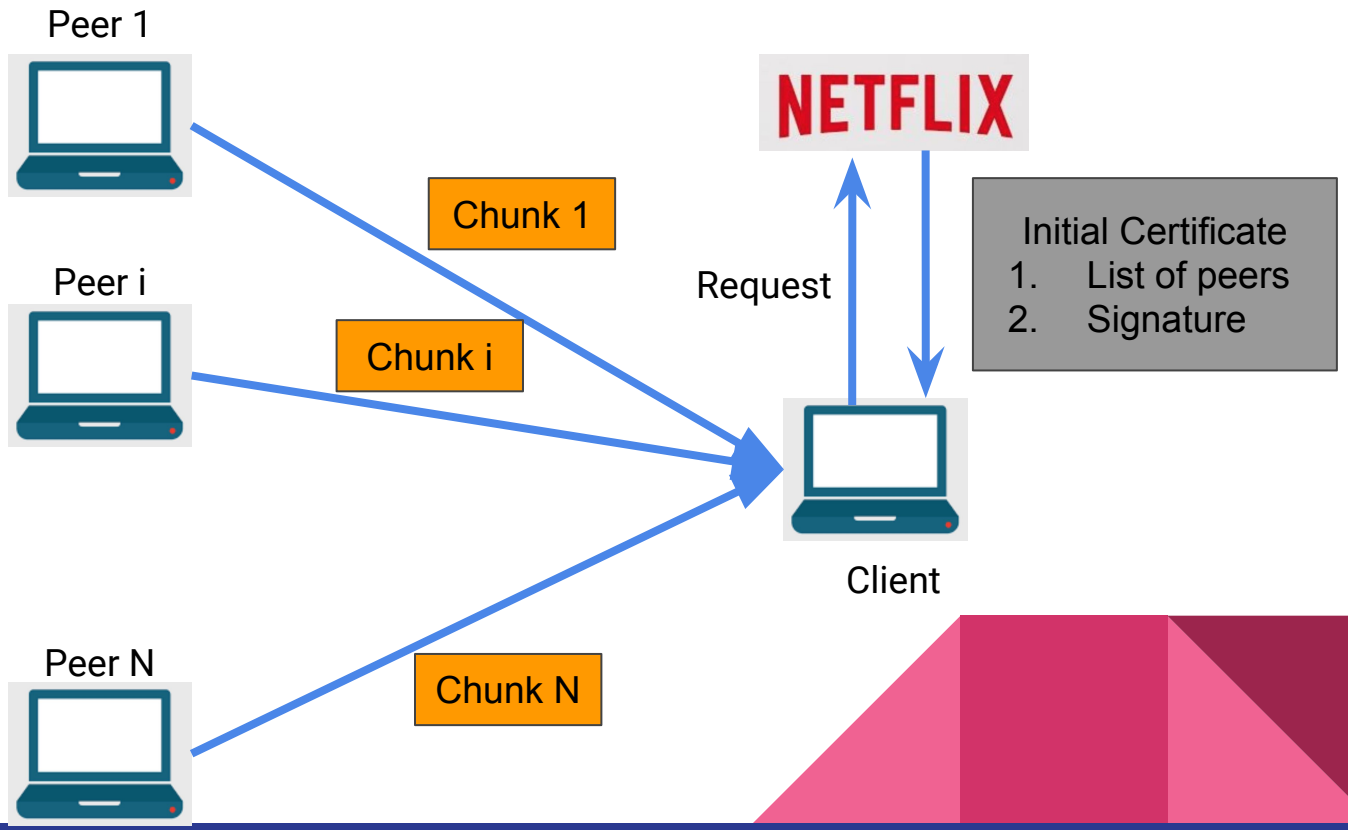


Design

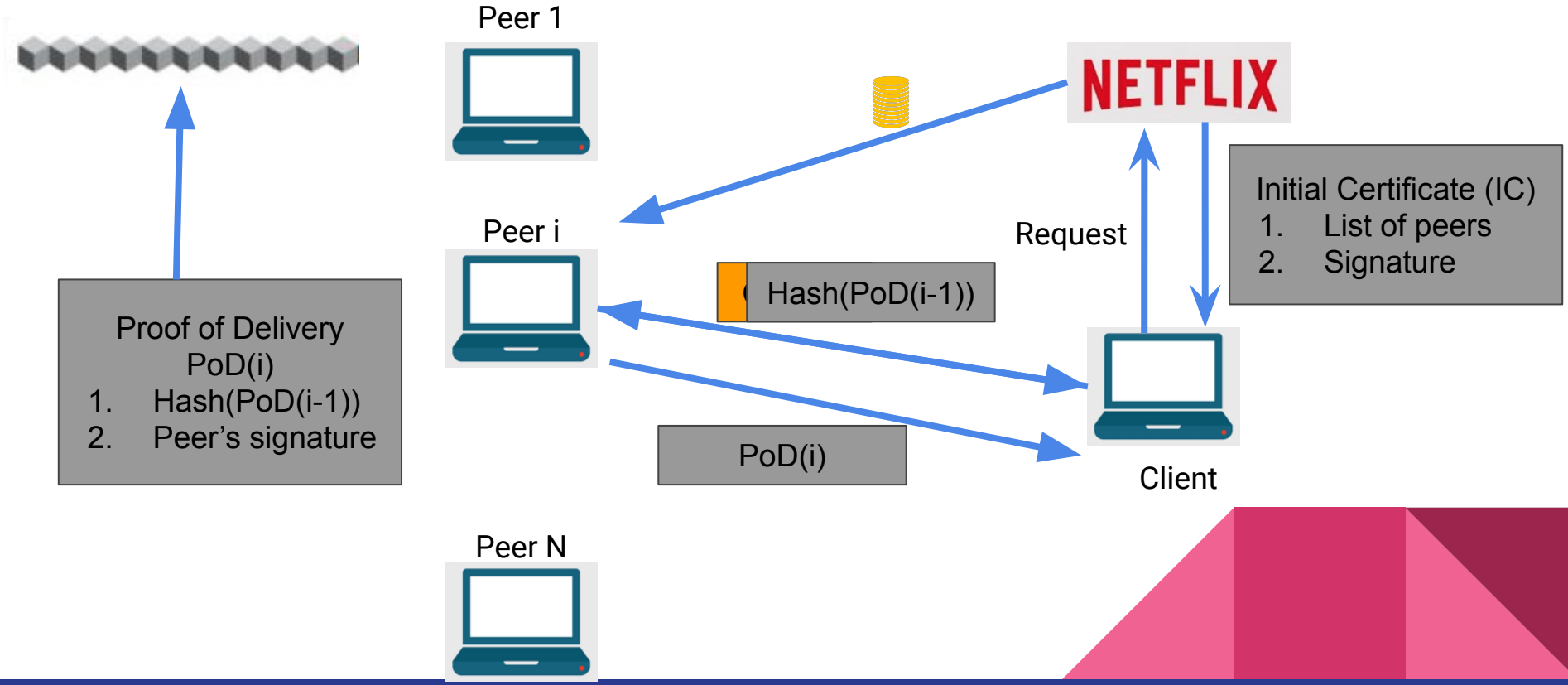

NETFLIX
Smart contract
1.Payment source
2.Defines PoD
Blockchain


Content file
Chunk 1
Chunk 2
Chunk N
Peers

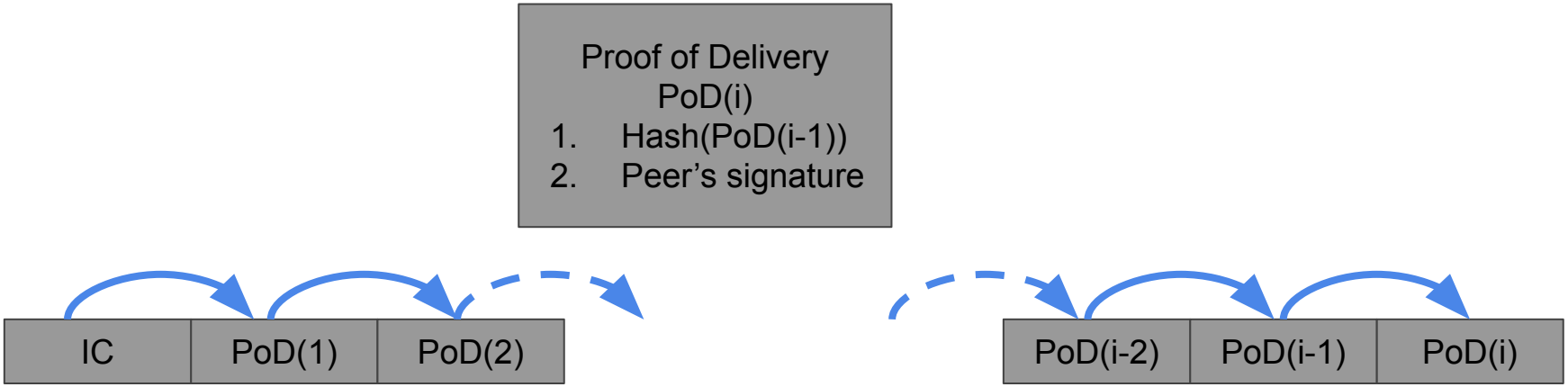
Downloading a file with Gringotts



Downloading a file with Gringotts



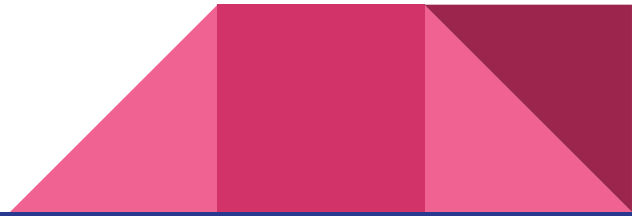
Proof of Delivery Chain (PoDC)



Clients should download chunks sequentially

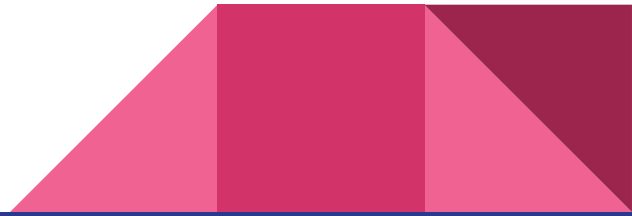
Detecting and thwarting attacks

- Content provider refuses to pay peer
- Peer earns money without delivering content



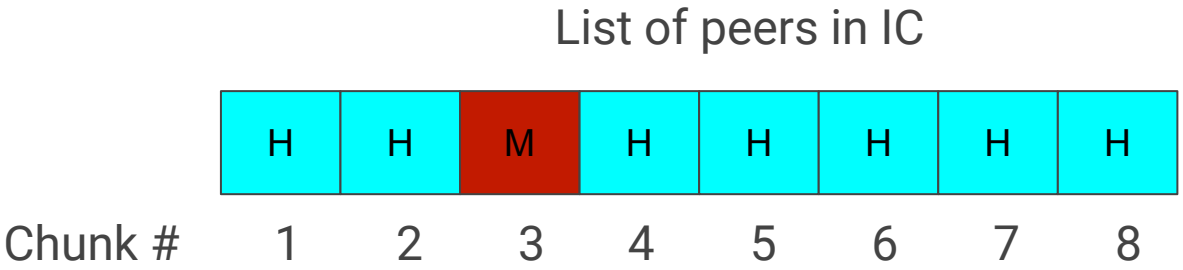
Collusion between clients and peers (Sybil)

- Attack: Clients and peers collude to generate PoD without serving any content



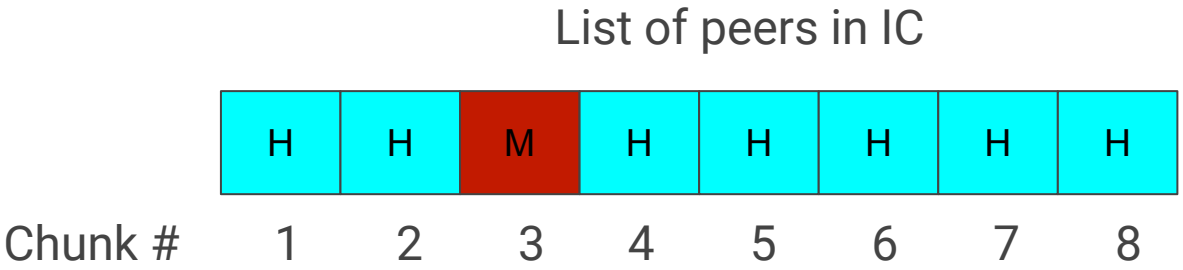
Collusion between clients and peers (Sybil)

- Attack: Clients and peers collude to generate PoD without serving any content



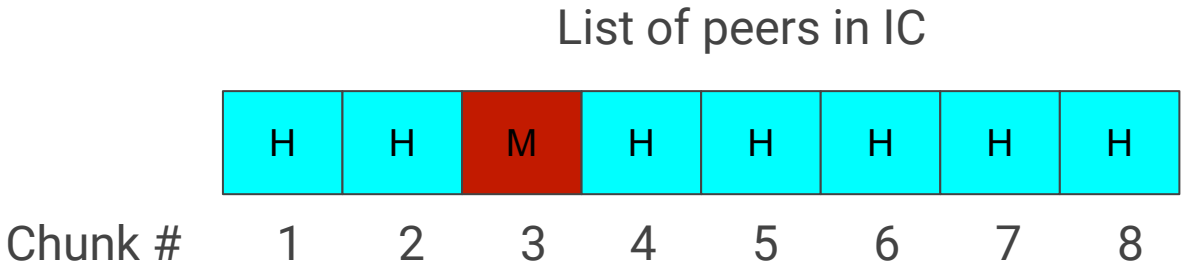
Collusion between clients and peers (Sybil)

- Attack: Clients and peers collude to generate PoD without serving any content
- Solution: Make attacks financially infeasible
  - PoD ensures that client downloads chunk sequentially
  - Random assignment of peers



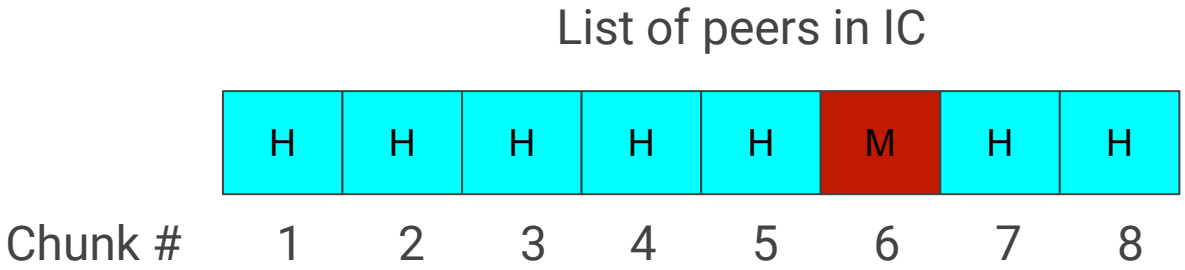
Collusion between clients and peers (Sybil)

- Attack: Clients and peers collude to generate PoD without serving any content
- Solution: Make attacks financially infeasible
  - PoD ensures that client downloads chunk sequentially
  - Random assignment of peers
- Model: Cost of creating client is 0 (Insufficient: Client can terminate connection)



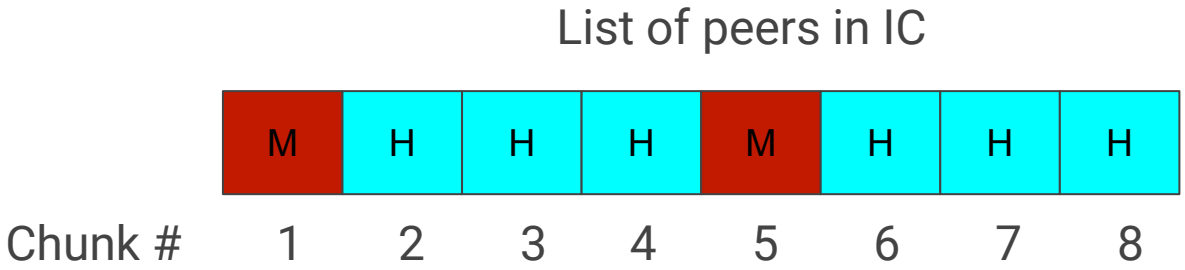
Collusion between clients and peers (Sybil)

- Attack: Clients and peers collude to generate PoD without serving any content
- Solution: Make attacks financially infeasible
  - PoD ensures that client downloads chunk sequentially
  - Random assignment of peers
- Model: Cost of creating client is 0 (Insufficient: Client can terminate connection)



Collusion between clients and peers (Sybil)

- Attack: Clients and peers collude to generate PoD without serving any content
- Solution: Make attacks financially infeasible
  - PoD ensures that client downloads chunk sequentially
  - Random assignment of peers
- Model: Cost of creating client is 0 (Insufficient: Client can terminate connection)



Collusion between clients and peers (Sybil)

Gringotts: Collusion is financially infeasible

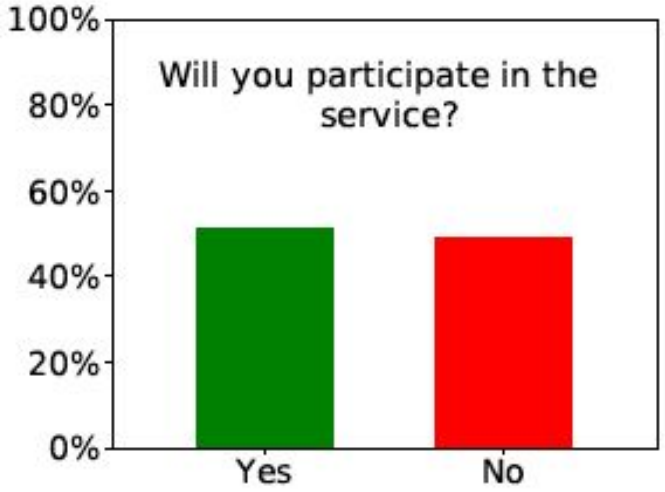
- Clients download first chunk from a secure node
- Overhead to client generation
  - Client must expend bandwidth resources before colluding with a peer

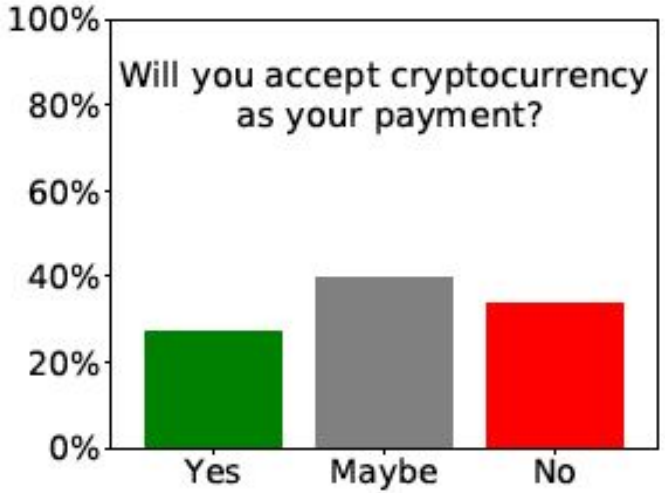


Consumer survey

- Understand expectations and requirements for peers
- 876 responders across USA
- Questions on
  - Payments
  - Resource availability
  - Participation concerns

Findings: Payments



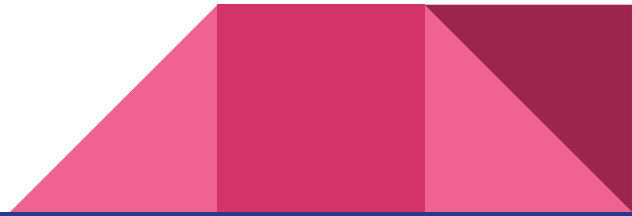


Findings: Concerns

Concern	Fraction concerned
Security and Privacy	82.8%
Liability for Illegal Content	50.5%
Performance Impact on Device	47.1%
Payment Concerns	42.1%
Personal Ethics	29.4%

Conclusion and discussion

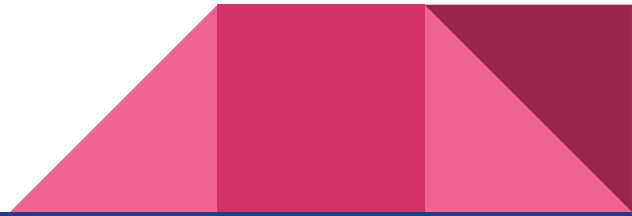
Gringotts: Secure monetary incentivization for decentralized content delivery



Conclusion and discussion

Gringotts: Secure monetary incentivization for decentralized content delivery

- Feedback
  - Additional challenges in P2P content delivery



Conclusion and discussion

Gringotts: Secure monetary incentivization for decentralized content delivery

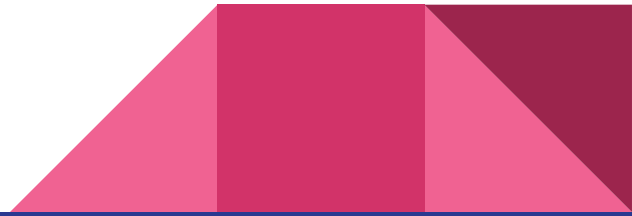
- Feedback
  - Additional challenges in P2P content delivery
- Controversial aspects
  - Gringotts increases upstream traffic and cost for ISPs
  - Use of cryptocurrency for payments



Conclusion and discussion

Gringotts: Secure monetary incentivization for decentralized content delivery

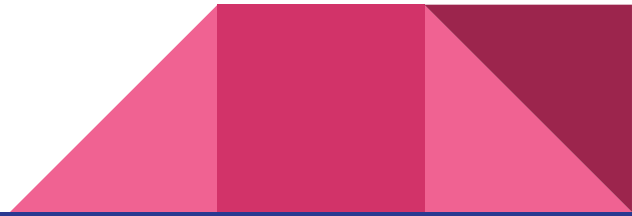
- Feedback
  - Additional challenges in P2P content delivery
- Controversial aspects
  - Gringotts increases upstream traffic and cost for ISPs
  - Use of cryptocurrency for payments
- Open issues
  - Consumer study (security and liability)
  - Ensuring QoS (file placement)



Conclusion and discussion

Gringotts: Secure monetary incentivization for decentralized content delivery

- Feedback
  - Additional challenges in P2P content delivery
- Controversial aspects
  - Gringotts increases upstream traffic and cost for ISPs
  - Use of cryptocurrency for payments
- Open issues
  - Consumer study (security and liability)
  - Ensuring QoS (file placement)
- Circumstance for failure
  - Failure to attract enough peers



Thank You

Detecting and thwarting attacks

- Single entity attacks
  - Malicious peer
  - Malicious content provider
- Collusion attacks
  - Collusion between clients and peers (Sybil attack)



Single entity attacks

- Malicious peer
  - Attack: Generate PoD without serving content
  - Peer requires PoDC, client can refuse PoDC if peer doesn’t serve content



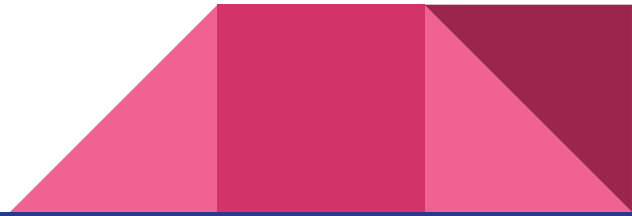
Single entity attacks

- Malicious peer
  - Attack: Generate PoD without serving content
  - Peer requires PoDC, client can refuse PoDC if peer doesn’t serve content
- Malicious content provider
  - Attack: Generate IC so that peers produce PoD not eligible for payments
  - Content provider can’t predict hash of PoD



Blockchain overheads

- Existing blockchains can only support 25 tx/sec
- Probabilistic payments to reduce overhead
  - Peers submit PoD to the blockchain, if Hash(PoD) % N = 0



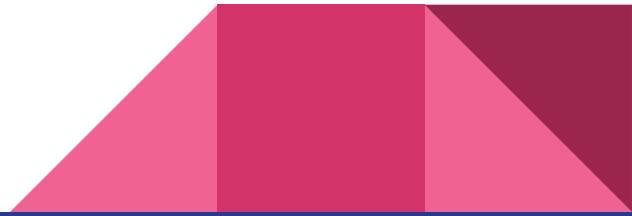
Single entity attacks

- Malicious peer
  - Attack: Generate PoD without serving content
  - Peer requires PoDC, client can refuse PoDC if peer doesn’t serve content
- Malicious content provider
  - Attack: Generate IC so that peers produce PoD not eligible for payments
  - Content provider can’t predict hash of PoD
  - Corner case: Content provider reusing IC
  - Include nonce in IC to avoid regenerating the IC



Collusion between clients and peers (Sybil)

- $x: Cient’s download cost per chunk
- $y: Price paid by content provider per chunk
- $z: Peer’s upload cost per chunk
- M: Fraction of malicious peers



Sybil attack: example

- $x: Client's download cost, $y: Price paid by content provider, $z: Peer's upload cost, m: fraction of malicious peers
- Expected cost of download upto chunk i

$$\texttt{Cost}[i] = 1 \cdot x + (i-1) \cdot (m \cdot 0 + (1-m) \cdot x)$$

Sybil attack: example

- $x: Client's download cost, $y: Price paid by content provider, $z: Peer's upload cost, m: fraction of malicious peers
- Expected cost of download upto chunk i

$$\texttt{Cost}[i] = 1 \cdot x + (i-1) \cdot (m \cdot 0 + (1-m) \cdot x)$$

- Expected payment to malicious peers

$$\texttt{Payment}[i] = (i-1) \cdot m \cdot y$$

Sybil attack: example

- $x: Client's download cost, $y: Price paid by content provider, $z: Peer's upload cost, m: fraction of malicious peers
- An attack is economically unfeasible if

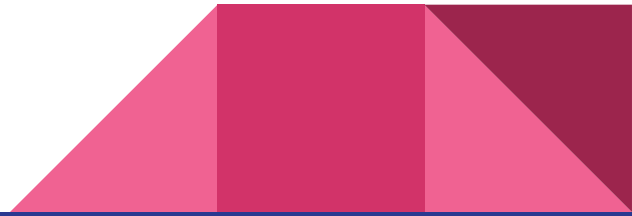
$$\texttt{Payment}[i] < \texttt{Cost}[i], \qquad \forall i \in \mathbb{N}$$

$$\Longrightarrow \qquad y \leq x\left(\frac{1}{m} - 1\right)$$

- E.g., If m=33%, then attack is economically infeasible if
  - y ≤ 2 . x (or payment ≤ 2 . Clients download cost)

Findings: Feasibility

- 82% users have more than 20GB free disk space
  - Store 100 videos
- Average uplink bandwidth: 22 Mbit/s
  - Stream 4 HD video streams simultaneously



Findings: Payment Concerns

Concern	Fraction concerned
Don't Know How to Use/Sell	59.7%
Not Setup to Receive	54.7%
Volatility and Risk	49.1%
Don't Know What it is	21.9%
Other	6.9%

Table 2: **Concerns for users who specified that they would *not* be willing to accept payment in cryptocurrency.**

Downloading a file with Gringotts

